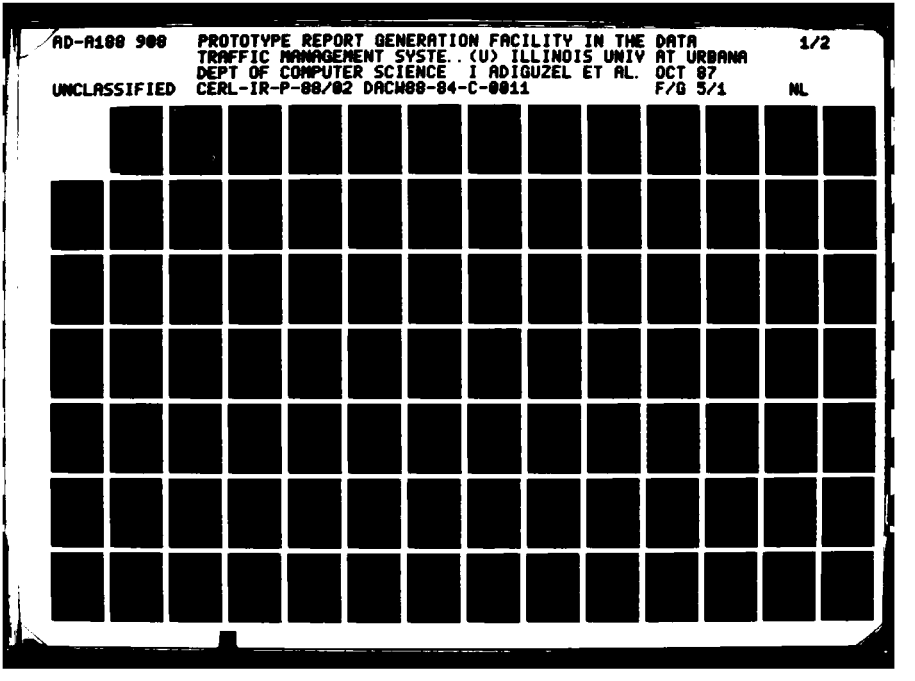
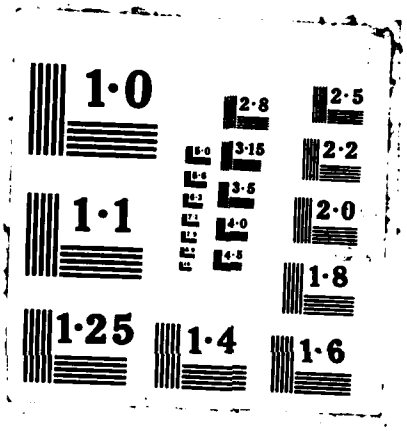
AD-A188 908 PROTOTYPE REPORT GENERATION FACILITY IN THE DATA TRAFFIC MANAGEMENT SYSTE.. (U) ILLINOIS UNIV AT URBANA DEPT OF COMPUTER SCIENCE I ADIGUZEL ET AL. OCT 87 1/2

UNCLASSIFIED CERL-IR-P-88/02 DACW88-84-C-0011 F/G 5/1 NL




1·0
1·1
1·25
1·4
1·6
1·8
2·0
2·2
2·5
2·8
3·15
3·5
4·0

US Army Corps of Engineers
Construction Engineering Research Laboratory

AD-A188 908

USA-CERL INTERIM REPORT P-88/02
October 1987
Responsiveness Analysis of Military Programs (RAMP)

Prototype Report Generation Facility in the Data Traffic Management System

by
Ilker Adiguzel
Geneva G. Belford
Jane W-S. Liu

The Data Traffic Management System (DTMS) is being developed to support mutual data consistency among a large number of military construction automated systems and coherent access to information stored on the systems. A prototype Report Generation Facility (RGF), a subsystem of the DTMS, has been designed to provide the coherent access to data systems served by the DTMS. Implementation completed to date on the prototype RGF is described.

The RGF consists of three major components: a Data Dictionary/Directory System (DDS), a User Interface (UI), and a Data Systems Interface (DSI). Currently, the prototype RGF is linked to two data systems—Construction Appropriations, Programming, Control and Execution System (CAPCES), and Military and Civil Progress Reporting System (MCPRS)—and supports only standard report identification, retrieval, and execution.

The computer program described herein is furnished by the Government and is accepted and used by the recipient with the express understanding that the United States Government makes no warranties, express or implied, concerning the accuracy, completeness, reliability, useability, or suitability for any particular purpose of the information and data contained in this program or furnished in connection therewith, and the United States shall be under no liability whatsoever to any person by reason of any use made thereof. The program belongs to the Government. Therefore, the recipient further agrees not to assert any proprietary rights therein or to represent this program to anyone as other than a Government program.

NORMAN C. HINTZ, AIA, PE
Colonel, Corps of Engineers
Commander and Director

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No 0704 0188
Exp Date Jun 30 1986

1a REPORT SECURITY CLASSIFICATION
UNCLASSIFIED

1b RESTRICTIVE MARKINGS

2a SECURITY CLASSIFICATION AUTHORITY

2b DECLASSIFICATION / DOWNGRADING SCHEDULE

3 DISTRIBUTION / AVAILABILITY OF REPORT
Approved for public release; distribution is unlimited.

4 PERFORMING ORGANIZATION REPORT NUMBER(S)
~~USA~~-CERL IR-P-88/02

5 MONITORING ORGANIZATION REPORT NUMBER(S)

6a NAME OF PERFORMING ORGANIZATION
U.S. Army Construction Engr Research Laboratory

6b OFFICE SYMBOL (If applicable)

7a NAME OF MONITORING ORGANIZATION

6c ADDRESS (City, State, and ZIP Code)
20 Massachusetts Ave., N.W.
Washington, D.C. 20314-1000

7b ADDRESS (City, State and ZIP Code)

8a NAME OF FUNDING / SPONSORING ORGANIZATION
HQUSACE

8b OFFICE SYMBOL (If applicable)
CEEC-P

9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
OMA project Responsiveness Analysis of Military Programs (RAMP)

8c ADDRESS (City, State, and ZIP Code)
20 Massachusetts Ave., N.W.
Washington, D.C. 20314-1000

10 SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO	PROJECT NO	TASK NO	WORK UNIT ACCESSION NO

11 TITLE (Include Security Classification)
Prototype Report Generation Facility in the Data Traffic Management System (Unclassified)

12 PERSONAL AUTHOR(S)
Adiguzel, Ilker; Belford, Geneva G.; Liu, Jane W-S.

13a TYPE OF REPORT	13b TIME COVERED	14 DATE OF REPORT (Year, Month, Day)	15 PAGE COUNT
Interim	FROM _____ TO _____	October 1987	116

16 SUPPLEMENTARY NOTATION
Copis are available from the National Technical Information Service Springfield, VA 22161

17 COSATI CODES

FIELD	GROUP	SUB GROUP
05	01	

18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
Data Traffic Management System military construction
Report Generation Facility
management information systems

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

The Data Traffic Management System (DTMS) is being developed to support mutual data consistency among a large number of military construction automated systems and coherent access to information stored on the systems. A prototype Report Generation Facility (RGF), a subsystem of the DTMS, has been designed to provide the coherent access to data systems served by the DTMS. Implementation completed to date on the prototype RGF is described.

The RGF consists of three major components: a Data Dictionary/Directory System (DDS), a User Interface (UI), and a Data Systems Interface (DSI). The DDS maintains all location and access path information on data items stored in data systems served by the RGF. The UI is designed to accept an English-like query; one of its major functions is to determine if the user query can be answered by using one or more of the standard reports

(Cont'd)

20 DISTRIBUTION / AVAILABILITY OF ABSTRACT
☐ UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT ☐ DTIC USERS

21 ABSTRACT SECURITY CLASSIFICATION
UNCLASSIFIED

22a NAME OF RESPONSIBLE INDIVIDUAL
D.F. Finney

22b TELEPHONE (Include Area Code)
(217) 352-6511 (x389)

22c OFFICE SYMBOL
CECER-IMT

DD FORM 1473, 84 MAR

83 APR edition may be used until exhausted
All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

BLOCK 19 (Cont'd)

provided by the individual data systems. When users request information contained in standard reports, the UI informs them of these reports' existence and allows them to select one or more of the reports. The keywords and qualifications provided in a user's query are passed to the DSI, which then links the RGF with each data system served by the DTMS.

When a standard report is requested, the DSI identifies and invokes the report generation routine for that report and produces a report. A nonstandard report generator will be provided in future RGF versions. The nonstandard report generator will be designed to assist the user in searching and accessing data systems and formatting the retrieved information. It will be invoked when no standard report is adequate to satisfy the user's query. Currently, the prototype RGF is linked to two data systems--Construction Appropriations, Programming, Control and Execution System (CAPCES), and Military and Civil Progress Reporting System (MCPRS)--and supports only standard report identification, retrieval, and execution.

FOREWORD

This investigation was performed for the Directorate of Engineering and Construction, Headquarters, U.S. Army Corps of Engineers (HQUSACE), under the Operations and Maintenance, Army (OMA) project "Responsiveness Analysis of Military Programs (RAMP)." The HQUSACE Technical Monitor was John J. Sheehey III, CEEC-P.

The work was performed by the Facility Systems Division (FS) of the U.S. Army Construction Engineering Research Laboratory (USA-CERL) in conjunction with the University of Illinois (U of I), Department of Computer Science (Contract Number DACW88-84-C-0011). Other USA-CERL personnel involved in the study were Roger L. Lapp and Gregory L. Robinson. E. A. Lotz is Chief, FS. The technical editor was Dana Finney, USA-CERL Information Management Office.

Other members of the U of I staff contributing to the project included Steven C. Hwung, Kenneth A. Kaufman, Chong Kwon Kim, James Leo, Debora Neff, Sudha Ram, Kenneth Rossen, Yi-Ling Yan, Longxiang Zhang, and Nancy Zyer.

COL Norman C. Hintz is Commander and Director of USA-CERL, and Dr. L. R. Shaffer is Technical Director.

CONTENTS

TABLES

FIGURES

PROTOTYPE REPORT GENERATION FACILITY IN THE DATA TRAFFIC MANAGEMENT SYSTEM

1 INTRODUCTION

Background

Information on U.S. Army military construction (MILCON) projects is stored on a large number of different major data systems Army-wide. Each system serves a set of special needs for the Army branch using it; however, different hardware and software configurations often impede data transfer and consistency among systems, frustrating attempts to access all relevant information.

The U.S. Army Construction Engineering Research Laboratory (USA-CERL) is developing the Data Traffic Management System (DTMS) to support mutual data consistency among MILCON data systems and coherent access to information stored in these systems. The DTMS is a collection of software facilities currently residing on a McDonnell Douglas Information Support Group (formerly TYMSHARE) computer.

The DTMS consistency function involves automatically capturing data updates in one system and posting them to other systems (regardless of hardware and software configurations) with no human intervention.[1] The DTMS coherent access function solves two related problems--retrieval of standard reports and generation of ad hoc reports from multiple data systems. USA-CERL, in conjunction with the University of Illinois, is developing the Report Generation Facility (RGF), a DTMS subsystem to handle the coherent access function.

The RGF provides standardized ways to produce reports with minimum user expertise and supports location and retrieval of data from many sources to produce integrated reports. The RGF consists of three major components: a Data Dictionary/ Directory System (DDS), a User Interface (UI), and a Data Systems Interface (DSI). The DDS maintains all location and access path information on data items stored in data systems served by the RGF. The UI is designed to accept an English-like query. It processes the query to determine which data system(s) contains the information requested by the user. One of the UI's major functions is to determine if a user query can be answered by using one or more of the standard reports provided by the individual data systems. When the user requests information contained in standard reports, the UI announces the existence of these reports and allows the user to select one or more of them. The keywords and qualifications provided in the user's query are passed to the DSI, which then links the RGF with each of the data systems served by the DTMS. When a standard report is requested, the DSI identifies and invokes the report generation routine of that report and produces a report. A nonstandard report generator will be provided in future RGF versions. The nonstandard report generator will be designed to assist the user in searching and accessing the data systems and formatting the retrieved information. It will be invoked when no standard report can satisfy the user's query.

---

[1] G. G. Belford, et al., "Mutual Consistency Maintenance in a Prototype Data Traffic Management System," *Proceedings of IEEE COMPDEC* (Institute of Electrical and Electronics Engineers, April 1984).

The RGF is a powerful, automatic system that can generate different types of reports containing data from the systems it serves. The prototype RGF is linked to two representative Army data systems, the Construction Appropriations, Programming, Control, and Execution System (CAPCES), and Military and Civil Progress Reporting System (MCPRS), and supports only standard report identification, retrieval, and execution. However, the prototype RGF can be easily expanded to link with the other data systems. In addition, the capability to generate nonstandard reports in the RGF will be included in the near future.

Purpose

The overall purpose of this work was to develop an automated system that will ensure mutual consistency among and coherent access to information stored in the Army's major data systems. The purpose of this interim report is to describe the RGF--the coherent access component of the overall system--in terms of structure, function, and implementation work to date.

Approach

Commercially available reports generation systems were investigated for applicability to the DTMS. The DDS selected is DATAMANAGER, from MSP, Inc.

The current RGF is designed to run under the IBM Conversational Monitoring System (CMS) environment and interface with the FOCUS data base management system. All major programs in the RGF are written in PL/1, making it easily portable in an IBM environment. (The Programming, Administration, and eXecution [PAX] System is the DTMS environment, currently using an IBM 370/3083 host computer. The PAX environment is expected to use IBM-type computers in the future.)

Scope

This report describes the RGF and implementation work completed to date. Future versions of the RGF will be described in reports that will supersede this one.

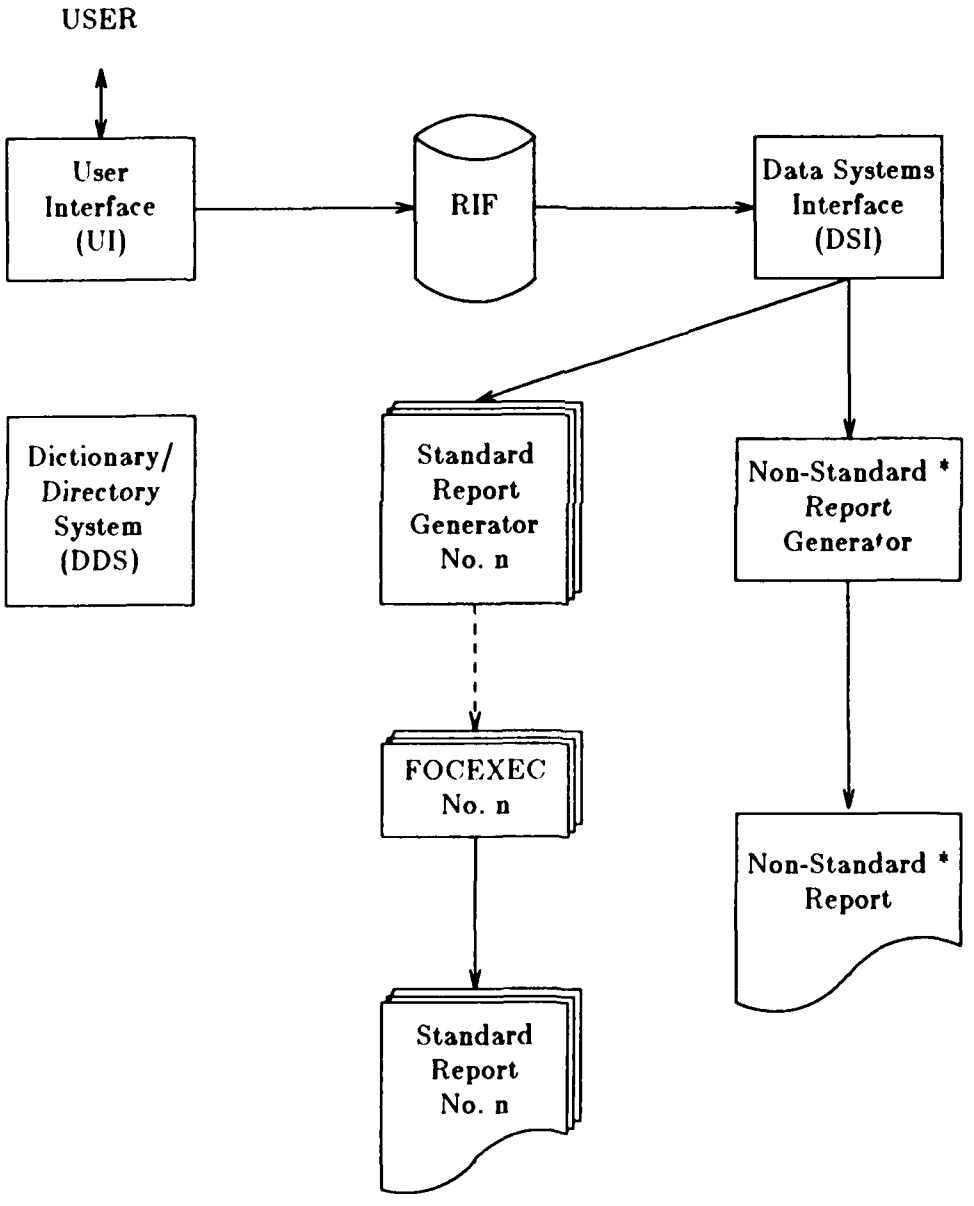
2 RGF STRUCTURE AND FUNCTIONS

Figure 1 is a schematic diagram of the RGF. Metadata (i.e., information on data stored in the data systems served by the RGF) are kept in the DDS (DATAMANAGER). Specifically, the DDS provides location and access path information and lends support to the DSI for access to individual data systems. In addition, the DDS supports keyword searches and maintains descriptive information on standard report generators linked to the DSI.

The UI is designed to accept an English-like query. To allow the user to phrase a query using familiar terms, the UI maintains a Thesaurus of Search Keywords and Internal Keywords (TSK). The TSK maps user-defined keywords to official internal keywords maintained in the DDS. The UI analyzes the query, captures the search keywords by consulting the TSK, and accesses the DDS to retrieve the necessary information on target data items, such as the item names, format, and location. The data item names and qualifications thus obtained by the UI are placed in the Report Information File (RIF). Based on information in the RIF, the UI identifies the data system(s) containing the data requested by the user. Another major function of the UI is to determine if the user query can be answered by one or more standard reports provided by the individual data systems or by the RGF. As mentioned in Chapter 1, when the user requests information contained in one or more standard reports, the UI announces the existence of these reports and allows the user to select one of them. Should this report not completely satisfy the user, a second report can be chosen, and so forth.

The DSI links the RGF with each data system served by the DTMS. The DSI interacts with the UI via the RIF. When a standard report is requested, the DSI identifies and invokes the report generation routine that produces it. This routine identifies target data items and qualifications of that report. (A target data item is an item accessed by the report generation routine. It forms a part of the report contents. A qualification is not used directly to generate the report; rather, it narrows the range of search by the report generation routine. For example, it might limit the report to data for a particular year or project.) If the user does not provide some qualifications needed for the report in the initial query, the selected report generation routine obtains them by prompting the user. After all qualifications are obtained, the data access and formatting routines are invoked to produce the report. When a standard report generation routine is already provided in the data system, the RGF makes use of that routine rather than duplicating it in the facility.

In the next RGF version, when no standard report can satisfy the query, the UI will allow the user to select the nonstandard report generator that will be provided. In this case, the nonstandard report generator will help the user to: determine which data systems hold the needed data, access those systems to retreive the data, and format the retrieved information.



* not implemented yet

Figure 1. The RGF and standard report generators.

3 USER INTERFACE AND QUERY PROCESSING

The UI is being implemented in the prototype RGF. Figure 2 shows the UI configuration. The UI is supported by the DDS, under which a Dictionary of Data Definitions (DDD) is maintained. The UI consists of two major components--the Query Parser (QP) and the Decision Maker (DM). The QP and the DM use the TSK and DDD. Information obtained when the UI processes the query is placed in the RIF. The RIF provides input to the DSI (described in Chapter 4). Important features to understand in query processing are the query language supported by the UI, the UI's two components, and the contents and structures of the TSK and DDD.

Query Language

The UI accepts from the user a wide range of queries phrased in something like natural language. In particular, the query language allows the user to phrase queries in terms of familiar search words and qualifications. At the same time, the language is easy to parse.

The important components in any query are search keywords (names of target data items) and qualifications (used to narrow the scope of the search). The qualifications are optional. (Later in the session with the RGF, if the user selects a standard report as the response, the RGF will prompt for all necessary qualifications if they have not been entered in the initial query.) The search terms and qualifications may be separated by spaces, special symbols, and meaningless words.

The syntax of the query language is defined formally in Figure 3. In this figure, REPORT, IS, ARE, AND, =, WITH, WHERE, FOR, and ',' are terminal symbols. A <special string> refers to a term recognizable by the QP. This term can be a search keyword. Here, an "internal keyword" is defined as a term stored in the DDD that may be used to search the data dictionary for names, locations, and similar information on actual data items in the data systems served by the RGF. Alternatively, the <special string> may be an unofficial term defined earlier by the user. The corresponding internal keywords would then be stored in the TSK. A term defined in the TSK is called a "search keyword." Blank <space>, literal <string>, and integer <digit number> have their usual meanings. <Junk> refers to any string that is neither a terminal symbol nor a term defined in the TSK.

Three examples of legal queries are:

TELL ME SOMETHING ABOUT PROJECT COST

SHOW THE PROJECT COST FOR PROJECT=B12 AND FISCAL YEAR=81

REPORT PROGRAM COST WITH PROJECT=B12 AND YEAR=81.

In each of these queries, the recognized search keywords are "project" and "cost." In the first query, everything else is parsed as <junk>. In the second query, the search keyword "cost" is followed by a qualification beginning with the qualifier "FOR" and specifying values for "PROJECT" and "FISCAL YEAR." The third query is similar to the second one.

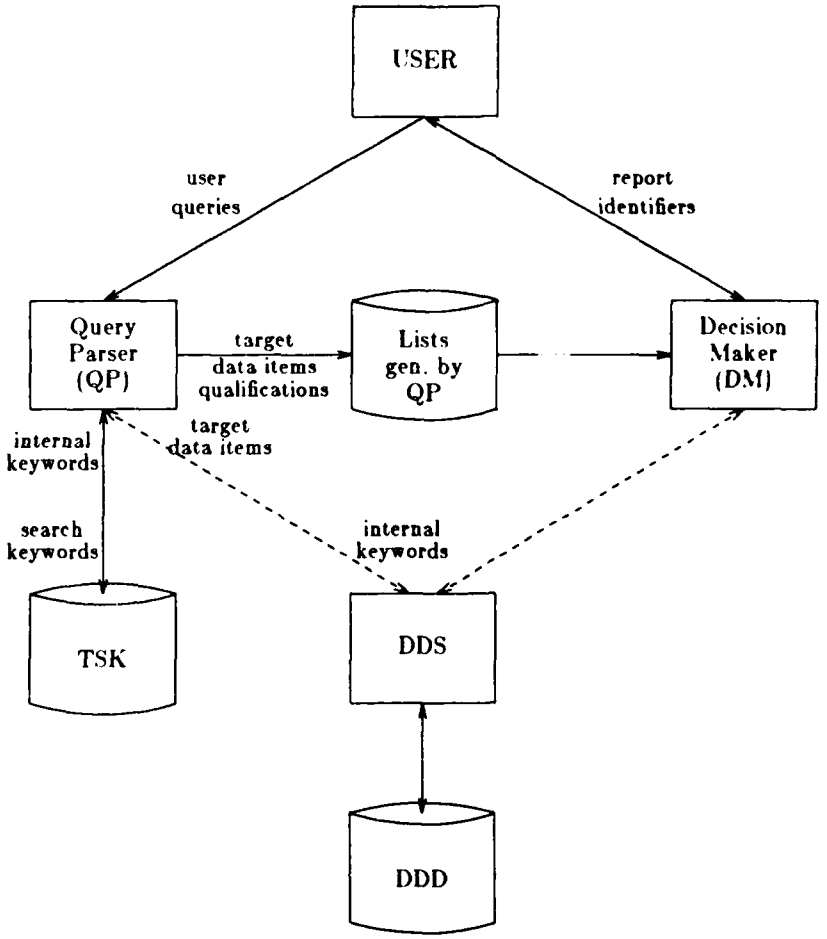


Figure 2. UI configuration.

```
<query statement>::=[REPORT]<target items>[<qualifier><qualification>]
<target items>::=<item><delim><target items>|<item>
<qualification>::=<item><equal><value>|
            <qualification><conjunction><item><equal><value>
<item>::=[<junk>]<item>[<junk>]<search keyword>[<junk>]|
                <search keyword>
<junk>::=<non-search keyword>|<junk><non-search keyword>
<qualifier>::=WITH|WHERE|FOR
<delim>::=,|AND
<conjunction>::=,|<space>|AND
<equal>::=IS|ARE|=
<value>::=<string>|<digit number>
<search keyword>::=<special string>
<non-search keyword>::=<string>
```

Figure 3. Query language syntax.

Thesaurus of Search Keywords (TSK)

To use a familiar term, the user first must define this term by entering the term's definition in the TSK. (When the term is not defined in the TSK, it is parsed as <junk>; i.e., it is not recognized by the QP.) Specifically, the definition of a term in the TSK is a mapping of it into a list of all corresponding "official" internal keywords and qualifying item names. Again, terms defined by the user in the TSK are search keywords; the TSK is designed to allow for natural expansion of the set of search keywords.

The TSK is implemented as a hash table that maps each search keyword to all corresponding internal keywords. The hash table consists of two parts: (1) a primary table implemented with a separate chaining and bucket-hashing scheme, and (2) an overflow table used to hold additional search keywords hashed to the same address. The overflow table has the same structure as the primary table, but the access method is different. Appendix A shows the hash table data structure.

For each term in the query, a hash address is generated using the hash function defined below where a term W of length k is represented as:

$$W = C_{k-1}C_{k-2}C_{k-3}\cdots C_2C_1C_0 \qquad \text{[Eq 1]}$$

Let S_i denote the converted internal bit-string for C_i. Then,

$$H(W) = (S_{k-1} * r^{k-1} + S_{k-2} * r^{k-2} + \cdots + S_1 * r^1 + S_0 * r^0) \bmod p \qquad \text{[Eq 2]}$$

In this implementation, p = 1013 and r = 7 (Appendix A explains p).

To locate the internal keyword list (if any exists) for a given term, the hash address is used as an index into the primary table. The term is then compared with the contents of the bucket entries. If the term is found in one of the records within the bucket, it is regarded as a search keyword and the corresponding internal keyword list comprises the mapped internal keywords for this search keyword. Otherwise, the overflow pointer is followed to find an overflow entry and the process is continued until either the term is found or no more entries exist. If the given term is not found in the hash table, it is not a search keyword; that is, it is not a term recognized by the QP.

Dictionary of Data Definitions (DDD)

The DDD is a facility managed by the DDS. It contains metadata about each item stored in the data system. That is, it contains item names, descriptions, formats, locations, and similar information. In addition, the DDD keeps descriptive information on all standard reports linked via the DSI to the RGF. That is, it contains information such as the report identifier, description, number and names of target data items, number of qualifications and the qualifying item names for each standard report.

The DDD is created and maintained under the DDS. It has a hierarchical structure with the following entities:

FOCUS-DATABASE

FOCUS-FILE

FILE

GROUP

ITEM.

Currently, the DDD contains metadata for the CAPCES and MCPRS data systems. Appendix B lists the CAPCES data elements as contained in the DDD.

The FOCUS-DATABASE entity (or record type) is at the top of the hierarchy. A data base, e.g., the CAPCES data base, can be a member of FOCUS-DATABASE. That is, any particular data base is an instance of the type FOCUS-DATABASE, and its descriptor would be entered as one of these top records in the hierarchy. Each FOCUS-DATABASE contains one or more FOCUS-FILEs. An example is the CAPCES PMMFILE. Its descriptor would therefore be entered as a FOCUS-FILE record under the CAPCES FOCUS-DATABASE record. In general, a FOCUS-FILE is a logically independent file unit in the FOCUS data base; it is the unit referred to in an OPEN or a CLOSE command, and contains one or more FILEs that correspond to the SEGMENTs in the FOCUS data base system. A GROUP in the FOCUS data base is described by the member type GROUP. An ITEM is an atomic entity in the DDD that describes a FIELD in the FOCUS data base system.

In the future, a REPORT entity will be added into the *DDD. This entity will* describe standard reports accessible via the RGF. Its attributes will include the report identifier, full report name, content description, number of target data items and their names, the number of qualifying items and their names, and similar information needed to complete the RIF. Currently, descriptive information on the standard reports is stored in the Standard Report Definition (SRDEF) File.

Internal keywords are contained in the CATALOGUE attribute for each ITEM member. The DDD can be accessed via the DATAMANAGER User Interface facility. The following two DATAMANAGER commands can be used to retrieve data in the SRDEF:

WHAT USES <item>

REPORT <item>.

The first command finds the access path for the item; the second one gets all attributes of the item.

Query Parser (QP)

The QP accepts user queries, picks up the search keywords by consulting the TSK, classifies each search keyword as either a target data item or a qualifying item, and associates each qualifying item with its value. It also finds the internal keywords corresponding to each search keyword in the query. The QP then accesses the DDD to get information about the real data items for target data and qualifying items.

In parsing a query, the QP first locates tokens in the query. A token can be a legal word, a '=', a ',', or a reserved word. A legal word in a query contains only letters, digits, and '_'. The reserved words in this query language include WITH, WHERE, FOR, IS, ARE, and AND. Figure 4 shows state transition for the QP.

The QP prompts the user to enter a query and parses the query according to the state transition diagram in Figure 4. It generates two lists--one containing the target data items and one containing the qualifying items found in the query. For each target data or qualifying item, more than one internal keyword may correspond to the search keyword used to name the item. The corresponding internal keyword list is retrieved from the TSK. After the QP translates the search keywords into internal keywords, it consults the DDD to get information about the real data item(s). This procedure helps detect nonexisting data items and reduces reporting errors.

The target items and qualifying items lists are organized as shown in Figures A2 and A3 in Appendix A. The DM uses these lists later.

Decision Maker (DM)

The DM accepts as inputs the lists of target data items and qualifying items generated by the QP. The DM consults the DDD to get relevant information regarding each item in these lists. (The UI facility of DATAMANAGER is used for this purpose.) This information is written into the RIF. The RIF consists of two subfiles--the report file and the data item file. Figure 5 shows their formats and contents. The RIF is the interface between the UI and DSI.

To determine how well each standard report matches the query, the following process is carried out: let the set of target data items in the query be $Q = \{q_1, q_2, ..., q_m\}$. Let R be the number of standard reports linked to the RGF. Each standard report i is characterized by the set $S_1 = \{3_1, 3_2, ..., 3_n\}$ of target data items contained in it, where n = f(i) is the number of target data items in the standard report i. The decision making process is then:

1. For i=1 to R, determine the degree of the match between the query and standard report i. Compare each target data item in Q with each target data item in S_i.

   - If $Q = S_i$, then standard report i satisfies the query exactly. The match factor is assigned the value 1.
   - If $Q\ S_i = \phi$, then standard report i does not satisfy the query. The match factor is assigned the value 3.
   - If $Q\ S_i \neq \phi$ and Q is contained in the set S_i, then standard report i satisfies the query but provides extraneous information. The match factor is assigned the value 1.
   - If $Q\ S_i \neq \phi$ and Q contains S, then standard report i meets the query in part. The match factor is assigned the value 2.

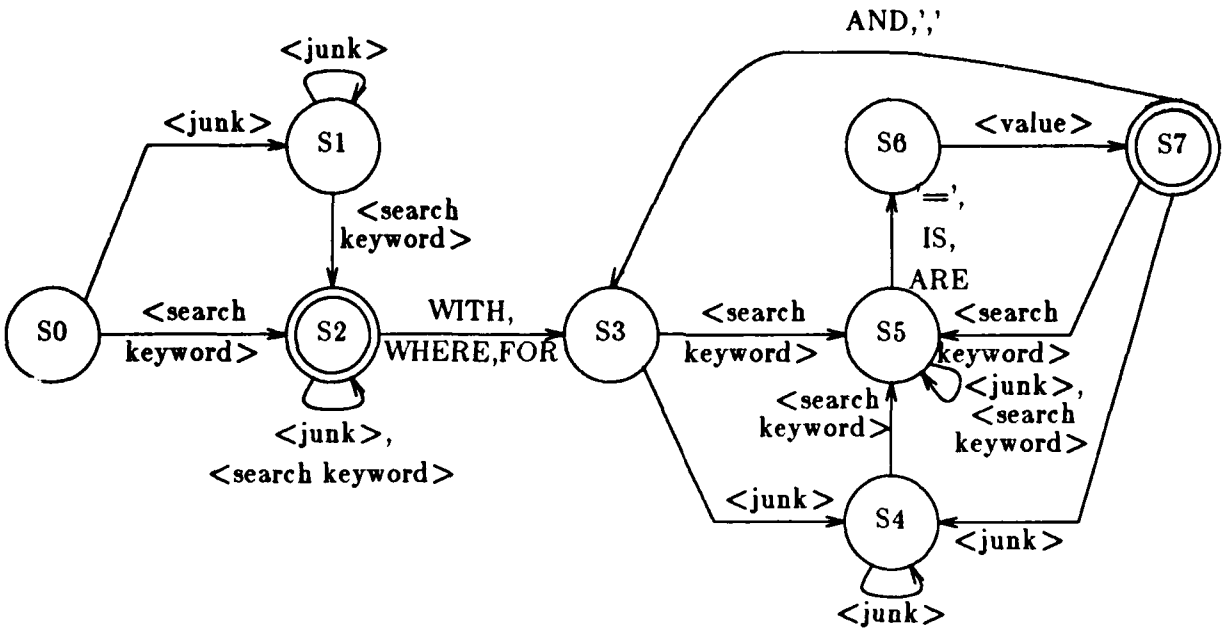


Figure 4. QP state transition.

Report File (RIFR)

Record 1	Report Identifier	Number of qualifications	Qualification				Qualification				...
			Name	Value	Index	Flag	Name	Value	Index	Flag	...
Format	I2	I2	X12	X12	I3	I1	X12	X12	I3	I1	...

Record 2	Number of target items	Index	Index	Index	Index	.....
Format	I2	I3	I3	I3	I3	.....

Data Item File (RIFD)

Record	Item Name	Format	Data-base	Logon ID	Passwd	Full Name	Item Description	Height	Access Path			
									seg	seg	seg	...
Format	X12	X10	X8	X8	X8	X32	X72	I2	X8	X8	X8	...

Figure 5. Formats of report file and data item file in RIF.

2. Count the number of reports with the same factor and store the count in the array pn of three components, one for each value of the match factor:

$$R = pn(1) + pn(2) + pn(3) \qquad \text{[Eq 3]}$$

where pn(j) is the number of reports that have match factors equal to j.

3. If pn(1) > 0, list the report identifiers and full names of the standard reports with match factors equal to 1 and ask the user if he or she wishes to select one of these reports. If the user finds one of the standard reports acceptable and selects it, set the SR flag and go to step 6.

4. If pn(2) > 0, list the report identifiers and full names of standard reports with match factors equal to 2; i.e., the ones that partly satisfy the query. If the user chooses one of them, set the SR flag and to go step 6. If the user finds none of these reports acceptable, he or she will be asked if a nonstandard report is required. If the answer is yes, set the NON_SR flag, and go to step 6. Otherwise, go back to the beginning and prompt the user for another query.

5. If pn(1) = pn(2) = 0, display the message: "no standard reports meet query requirements." Then ask the user whether a nonstandard report is desired. If the answer is yes, set the NON_SR flag and go to step 6.

6. Depending on whether the SR flag or the NON_SR flag has been set, write the query information to the RIF file.

After report identifiers and full names are shown on the screen, the user may enter "?" to request that descriptions for the standard reports be displayed. The descriptions allow a user to make an intelligent decision as to whether a given report fulfills all the requirements. When the user responds with a valid standard report identifier, the UI displays the selected report identifier and prompts the user to verify it. If the user enters "yes," the RIF file will be written and the report generation continues. Otherwise, the UI asks the user to reenter the report identifier.

4 DATA SYSTEMS INTERFACE

As mentioned earlier, DSI links the RGF to individual data systems served by the DTMS. The section of the DSI linking the RGF to a particular standard report generator is partitioned into two parts. The first part checks the RIF to determine if the user has provided all necessary qualifications in the initial query; if more qualifications are needed, it prompts the user to enter them. The second part invokes the data access and report generation routines to actually generate the report; these routines are executed by calling their names.

A PL/1 program named QUAL handles the first part. This program reads the two RIF files (RIFR and RIFD) (Figures 5 and 6) prepared by the UI.

The RIFR contains the report generation routine identifier along with target data and qualifying data items required by the routine. For each qualifying data item, two spaces are reserved in case a range of values is to be specified for the data item. Each data item has an associated flag that indicates whether a value has been specified for it. When the user has specified a value in the initial query, the flag is set to 1; otherwise, it is set to 0.

The RIFD contains detailed information about target data and qualifying data items of the chosen report generator routine. Specifically, it includes the format, location, full name, and description of each target data item and qualifying data item. This detailed information is displayed when the system prompts the user to specify values for the qualifications.

The program QUAL reads the flag for each qualifying data item sequentially. If a flag is 1, it passes to the next qualifying data item, provided there is one. If the flag is 0, the program prints the following message on the screen:

```
TO COMPLETE THE QUERY, YOU MUST SPECIFY
THE VALUES OF FOLLOWING DATA ITEMS:

   CFY  IN PMMFILE FORMAT=A2
   DESC:FISCAL YEAR.
```

The first two lines are printed only once--at the beginning of the procedure. The next two lines are repeated for each qualifying data item for which a value is as yet unspecified. Note that in the above message, information extracted from the RIF data file gives the user a more detailed explanation of the qualifying data items.

When a user replies to a prompt with a qualification value, the program accepts the value and inserts it into the corresponding position in the RIF. This process is repeated until all qualifications are specified. If all qualifications have been specified in the initial query phase, the user will not see the above message.

As mentioned earlier, the report generation routines already provided in the data systems are used when possible. Thus, new routines with the same function need not be written for the DSI. However, some modifications had to be made to the standard routines before including these routines in the DSI; this is because most of these routines interactively prompt the user to specify the qualifications needed to delineate a report

(but they do not help the user by giving detailed information such as the item's full name and description). The standard routines have been modified to read qualification values from the filled-in fields in the RIF files.

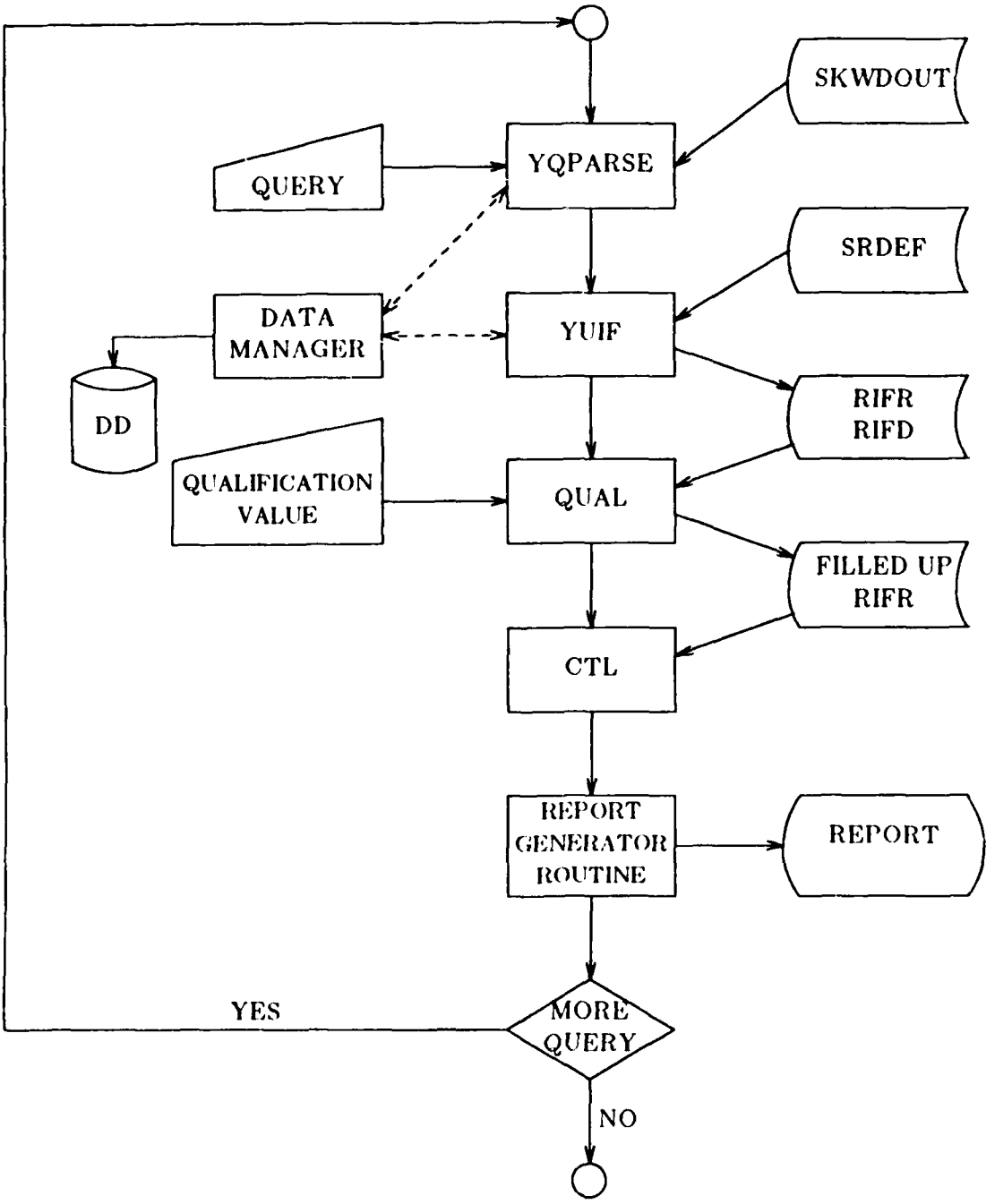


Figure 6. RGF flowchart.

5 CURRENT IMPLEMENTATION

The current prototype RGF is linked only to the CAPCES and MCPRS data systems. Six standard reports can be recognized and generated via the RGF. Appendix C shows the current contents of the system tables in the SRDEF that permit recognition of the first two reports. Additional standard reports can be integrated into the RGF by expanding these tables and modifying the DSI as indicated in Chapter 4. More data systems can be linked to the RGF by expanding the DDD, TSK, and tables in the SRDEF file.

RGF Operations: Example

Appendix D depicts a typical session of query processing. To illustrate the prototype RGF's operations, consider the query below:

TELL ME ABOUT COST SUMMARIZED BY PROJECT ELEMENT.

At the start of the session with the RGF, the SRDEF and TSK file are read into the RGF work area. The following steps are then carried out after the user types in the query:

1. The QP determines that COST, PROJECT, and ELEMENT are search keywords, since they are defined in the TSK (Table C3 in Appendix C).

2. From the TSK, the search keywords COST and ELEMENT are mapped into internal keywords AMOUNT and ELEMENT, respectively. The search keyword PROJECT is mapped into two internal keywords, PROJECT and PROGRAM.

3. The following two commands are issued to access DATAMANAGER and get the data item name, PROGELE:

WHAT FORMS'AMOUNT','PROJECT','ELEMENT'

WHAT FORMS'AMOUNT','PROGRAM','ELEMENT'

(Table C2 in Appendix C lists the data stored under DATAMANAGER. Only the data item PROGELE is indicated by internal keywords AMOUNT, PROGRAM, and ELEMENT.)

4. The data item name PROGELE is compared with the target data item names of the available standard reports. Using the DM procedure, standard report 1 is chosen. Finally, standard report 1 is generated by calling SUMELEM FOCEXEC.

Programs in the RGF

Figure 6 is a flowchart for the RGF. The functions of programs shown in this flowchart or listed in Appendix E are described below:

The program YSIMAP performs the following functions:

1. Maintains the TSK

2. Reads the data file SKWDIN

3. Generates the output file SKWDOUT.

The file SKWDOUT is used later in the program YQINIT for generating the hash table already described.

The program ZSRDEF has been designed to generate descriptive information on any standard report to be linked with the RGF and to enter the information in the SRDEF. Currently, this program prompts the user to enter the information needed to generate the standard report and records this information in the SRDEF. The information on a standard report contained in the SRDEF might also be generated by accessing that part of the DDD containing the descriptive information on the report. This part of ZSRDEF will be developed in the future.

The module YUSINTNEW implements the UI in the prototype RGF. It consists of seven programs:

1. YUSINIT integrates five programs, YQINIT, YAINIT, YUINIT, YQPARSE, and YUIF. YUSINIT calls YQINIT, YAINIT, and YUINIT to carry out the initializations. When the user enters a query, YUSINIT calls YQPARSE to phrase the query. It then calls YUIF to perform the decision procedure in the DM.

2. YQINIT initializes the program YQPARSE. Specifically, it reads the data file SKWDOUT to establish the hash table.

3. YAINIT initializes the program YACSD. It opens the DDD so it is ready for use.

4. YUINIT initializes the program YUIF. It reads the data file SRDEF and writes it to the local workspace in the module YUSINTNEW.

5. YQPARSE implements the query parser. To generate the lists of target data items and qualifying items, this program calls the program YACSD to access the DDD. The DATAMANAGER command WHAT FORMS is used to get the real data items.

6. YUIF implements the DM. It identifies the standard reports that match the user query and prompts the user to choose one of them. Then YUIF generates the RIF data files RIFR and RIFD, which are used later to generate the selected report.

7. YACSD accesses the DDD by calling the procedure DMRUS, which interfaces between the user's program and the DDS. YACSC issues the command WHAT FORMS with the parameters passed by YQPARSE and gets a list of real data items returned via the interface program DMRUS.

The QUAL program is responsible for completing the qualifications needed to generate the selected standard report. If there are qualifying data items with no values assigned in the RIF report file, the QUAL program prompts the user to specify values for them.

All the programs just mentioned are called from the CMS EXEC program STDREP. When the UI processes terminate and the user chooses a standard report, the STDREP calls the FOCUS EXEC program CTL, which invokes the needed report generation routine. After the selected standard report is generated, STDREP asks the user if

he or she wants to continue, that is, to type another query. Specifically, the following message is displayed:

SELECT ACTION TO BE TAKEN

1. EXECUTE A DIFFERENT QUERY.
2. EXIT FROM THE REPORT GENERATOR.

Appendix E includes all the PL/1 programs and the EXEC and DATA files used to implement report generation.

6 CONCLUSION

A prototype RGF has been developed to provide coherent access to all DTMS-served data systems Army-wide. It identifies standard reports that might satisfy the user query and, if the user wishes to have a standard report, automatically invokes the standard report generator to generate it. The prototype user interface provided in the RGF should enable the user to generate reports containing data retrieved from any number of data systems easily. The user need not have any technical knowledge about the data systems; queries can be issued using familiar terms with no concern for the location and method of data storage.

The DDS plays an important role in the UI, maintaining metadata about data in the data systems and thus allowing any data item queries by the user to be located and accessed readily. Supported by a well designed, powerful, commercially available dictionary/directory system, DATAMANAGER, the prototype RGF is readily expandable to link multiple data systems and numerous standard reports.

Future effort will be directed toward implementing the following enhancements:

1. More than one standard report may meet the user's requirements at any one time. A user will normally select one of them. After the selected report is generated, the user may want to generate one or more of the other reports that also match the query. A loop will be implemented to allow the user to request these additional standard reports.

2. The REPORT entity will be added into the directory so that the SRDEF file can be eliminated.

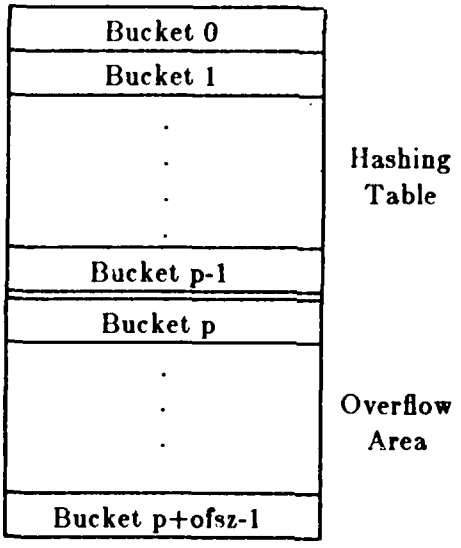
3. For every data system linked to the RGF, the contents of the CATALOGUE attribute in the DDD should be captured automatically from the data base definition. A program will be implemented in the DSI to scan the data base definition stored in any data system and load the DDD automatically. Thus, some aspects of the tedious loading task may be avoided.

4. The nonstandard report generator should be designed and a prototype nonstandard report generator implemented.

APPENDIX A:

HASH TABLE AND PROGRAM INTERFACE DATA STRUCTURES

The hash table is configured as follows:



where p = 1013 and ofsz (overflow table size) = 200 in this implementation.

Each bucket contains the information shown in Figure A1.

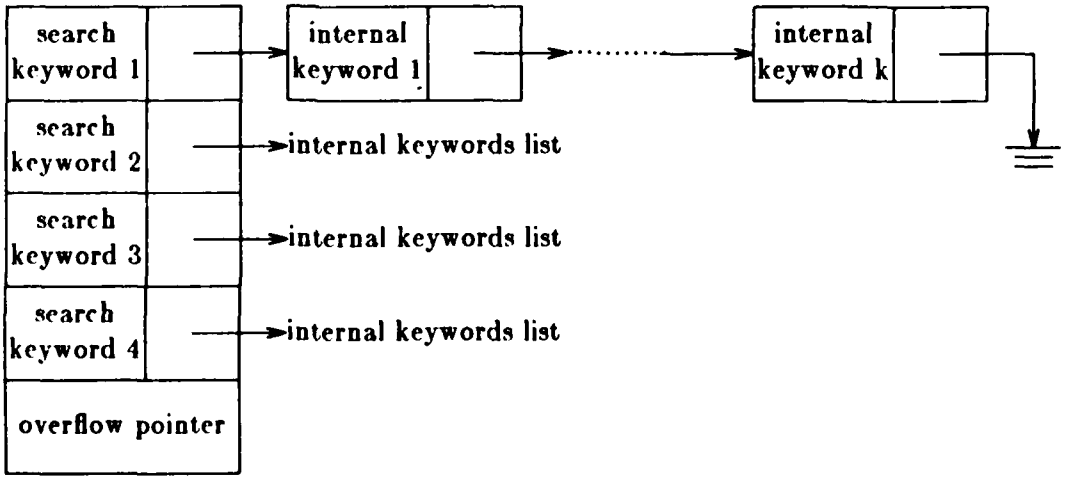


Figure A1. Data structure for the hash table.

The data structures used in the UI between the QP and DM for target data items and qualifying items are shown in Figures A2 and A3.

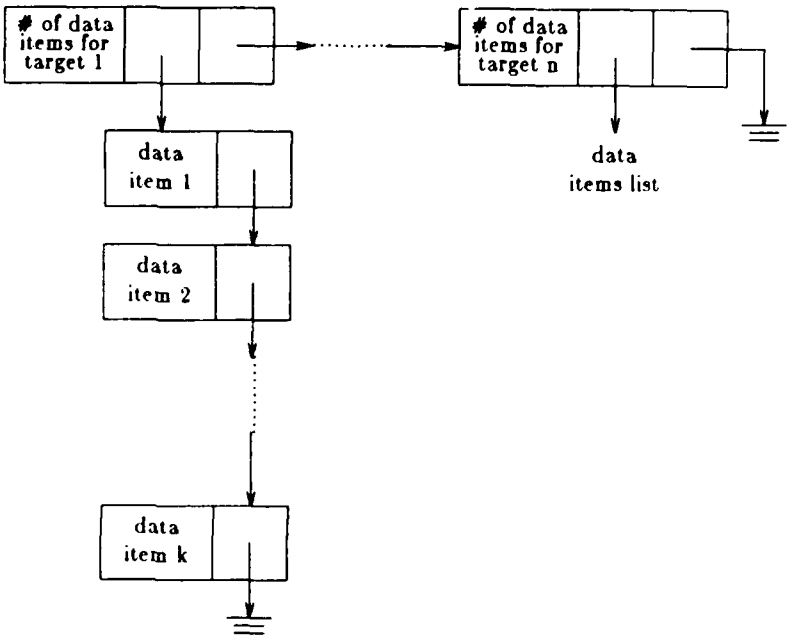


Figure A2. Data structures for target data items.

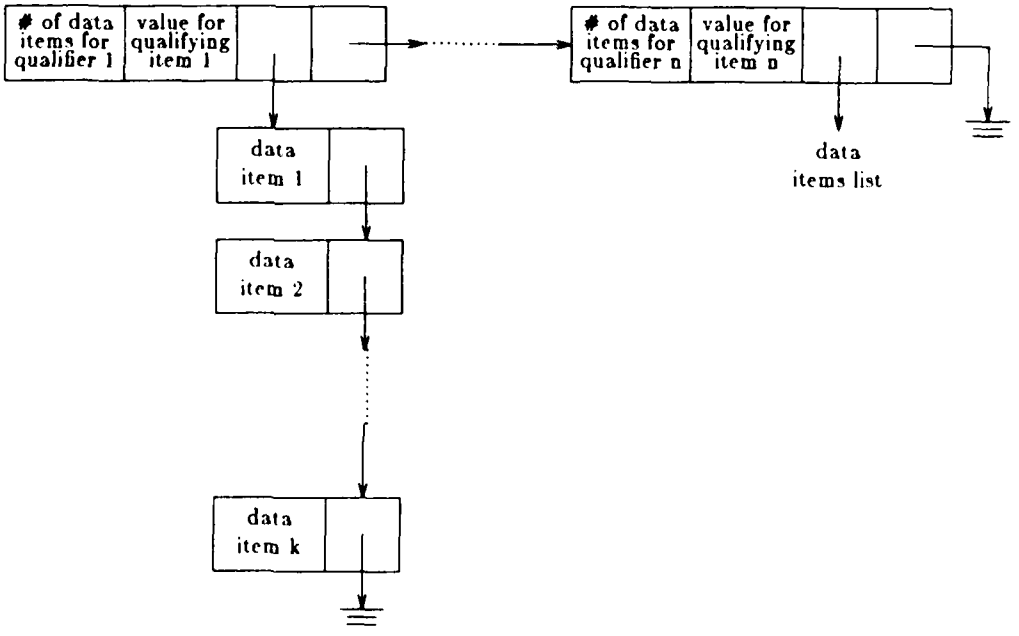


Figure A3. Data structures for qualifying items.

APPENDIX B:

CAPCES DATA IN THE DIRECTORY

```
C>yacsdmi terminal terminal
SET LDRTBLS 07
FILEDEF DMIN TERMINAL ( BLKSIZE 89 )
FILEDEF DMOUT TERMINAL ( BLKSIZE 89 )
FILEDEF DDTMS DISK CERL1 INDEX A ( XTENT 1000 )
FILEDEF DDTMSD DISK CERL1 DATA A ( XTENT 1000 )
FILEDEF DDTMSS DISK CERL1 SOURCE A ( XTENT 1000 )
FILEDEF DDTMSE DISK CERL1 RECOVER A ( XTENT 1000 )
FILEDEF DDTMSJ DISK CERL1 LOG A ( XTENT 1000 )
DM00
  PRODUCT DATAMANAGER     RELEASE 4.1.1      RELEASE DATE      20MAY83
  MACHINE IBM             MODEL   360        OPERATING SYSTEM CMS
  RELEASE TAPE  M63475    PRODUCED 14NOV83
  INSTALLATION
  UNIVERSITY OF ILLINOIS
>dictionary ddtms update;
DM01023I                  DDTMS     DICTIONARY OPEN
>authority 'uiucdcs';
DM01060I                  AUTHORITY RECOGNISED
>report pmmfile hierarchy;
REPORT OF FOCUS-FILE PMMFILE
  EDITION 1 ENCODED BY MASTER AT 19.28.36 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO 0 TIMES
  THIS MEMBER CONTAINS 38 DIRECT REFERENCES
  FOCUS-FILE PMMFILE
  FORM DEFAULTED-AS
  CONTAINS
    FILE  MAIN
    FILE  ZCPPFILE
    FILE  ZCP2FILE
    FILE  ZCP3FILE
    FILE  AMPERS1
    FILE  ACEFILE
    FILE  MPCAFILE
    FILE  MPESFILE
    FILE  MPENFILE
    FILE  SIGFILE
    FILE  REPRFILE
    FILE  SUBPFILE
    FILE  MACOMSEG
    FILE  HISTZCP1
    FILE  HISTZCP2
    FILE  HISTZCP3
    FILE  HISTZCP4
    FILE  HISTZCP5
    FILE  HISTACE1
    FILE  HISTMPES
    FILE  HISTMPE1
    FILE  HISTMPC1
    FILE  HISTMPC2
    FILE  HISTLVL
    FILE  HISTCPY
    FILE  COM2SEG
    FILE  SITESEG
    FILE  CATSEG
    FILE  CAT3DESC
    FILE  DESCRSEG
    FILE  DIVSEG
    FILE  INSTSEG
    FILE  MACMSEG
    FILE  ZBDICSEG
    FILE  AUTHSEG
    FILE  NOTEAUTH
    FILE  APPRSEG
    FILE  NOTESEG
REPORT OF FILE MAIN
  EDITION 1 ENCODED BY MASTER AT 19.23.18 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 28 DIRECT REFERENCES
  FILE MAIN
  SEGTYPE S
  SORT-KEY
    ITEM  KEYNR
  FORM DEFAULTED-AS
```

```
  CONTAINS
    ITEM  KEYNR
    ITEM  KN_UPD
    ITEM  KN_SDT
    ITEM  INST
    ITEM  PFT
    ITEM  FY
    ITEM  DATECFY
    ITEM  PCFY
    ITEM  OFY
    ITEM  CMDC
    ITEM  PRCD
    ITEM  ORIG_USVC
    ITEM  SITE_CODE
    ITEM  CATCDS
    ITEM  CMD_PRI
    ITEM  DD_SORT_CD
    ITEM  PROG_ELE
    ITEM  PROJECT_DESC
    ITEM  CURR_SCOPE
    ITEM  ORI_SCOPE
    ITEM  TYPE_FUNDS
    ITEM  MISSION
    ITEM  TEMP_PN
    ITEM  PERM_PN
    ITEM  MAIN_FILLER
    ITEM  RKEY
    ITEM  MOB_GROUP
REPORT OF ITEM KEYNR
  EDITION 1 ENCODED BY MASTER AT 19.22.58 ON 27 NOV 1984
  ALIAS
    KN
  THIS MEMBER IS DIRECTLY REFERRED TO 2 TIMES
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM KEYNR
  DEFAULTED-AS
    CHARACTERS 13
REPORT OF ITEM KN_UPD
  EDITION 1 ENCODED BY MASTER AT 19.23.00 ON 27 NOV 1984
  ALIAS
    KNU
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM KN_UPD
  DEFAULTED-AS
    CHARACTERS 1
REPORT OF ITEM KN_SDT
  EDITION 1 ENCODED BY MASTER AT 19.23.01 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM KN_SDT
  DEFAULTED-AS
    CHARACTERS 12
REPORT OF ITEM INST
  EDITION 1 ENCODED BY MASTER AT 19.23.01 ON 27 NOV 1984
  ALIAS
    STA_CD
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM INST
  DEFAULTED-AS
    CHARACTERS 5
REPORT OF ITEM PFT
  EDITION 1 ENCODED BY MASTER AT 19.23.02 ON 27 NOV 1984
  ALIAS
    PM_FILETYPE
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM PFT
  DEFAULTED-AS
    CHARACTERS 1
REPORT OF ITEM FY
  EDITION 2 ENCODED BY MASTER AT 15.00.38 ON 05 DEC 1984
  ALIAS
    CFY
  CATALOGUED AS
    CURRENT
    YEAR
    CONGRESS
    FISCAL
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
```

```
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM FY
   DEFAULTED-AS
      CHARACTERS 2
REPORT OF ITEM DATECFY
   EDITION 1 ENCODED BY MASTER AT 19.23.04 ON 27 NOV 1984
   ALIAS
      DT_CFY_CHG
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM DATECFY
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM PCFY
   EDITION 1 ENCODED BY MASTER AT 19.23.05 ON 27 NOV 1984
   ALIAS
      PREVIOUS_CFY
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM PCFY
   DEFAULTED-AS
      CHARACTERS 2
REPORT OF ITEM OFY
   EDITION 1 ENCODED BY MASTER AT 19.23.05 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM OFY
   DEFAULTED-AS
      CHARACTERS 2
REPORT OF ITEM CMDC
   EDITION 1 ENCODED BY MASTER AT 19.23.06 ON 27 NOV 1984
   ALIAS
      CMD_CD
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM CMDC
   DEFAULTED-AS
      CHARACTERS 2
REPORT OF ITEM PRCD
   EDITION 1 ENCODED BY MASTER AT 19.23.07 ON 27 NOV 1984
   ALIAS
      PROG_CD
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM PRCD
   DEFAULTED-AS
      CHARACTERS 2
REPORT OF ITEM ORIG_USVC
   EDITION 1 ENCODED BY MASTER AT 19.23.07 ON 27 NOV 1984
   ALIAS
      ORUSVC
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM ORIG_USVC
   DEFAULTED-AS
      CHARACTERS 2
REPORT OF ITEM SITE_CODE
   EDITION 1 ENCODED BY MASTER AT 19.23.08 ON 27 NOV 1984
   ALIAS
      INCODE
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM SITE_CODE
   DEFAULTED-AS
      CHARACTERS 5
REPORT OF ITEM CATCD5
   EDITION 1 ENCODED BY MASTER AT 19.23.09 ON 27 NOV 1984
   ALIAS
      CAT5
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM CATCD5
   DEFAULTED-AS
      CHARACTERS 5
REPORT OF ITEM CMD_PRI
   EDITION 1 ENCODED BY MASTER AT 19.23.09 ON 27 NOV 1984
   ALIAS
      CMDP
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM CMD_PRI
```

```
   DEFAULTED-AS
     CHARACTERS 5
REPORT OF ITEM DD_SORT_CD
   EDITION 1 ENCODED BY MASTER AT 19.23.10 ON 27 NOV 1984
   ALIAS
     DDSC
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM DD_SORT_CD
   DEFAULTED-AS
     CHARACTERS 2
REPORT OF ITEM PROG_ELE
   EDITION 2 ENCODED BY MASTER AT 14.07.44 ON 05 DEC 1984
   ALIAS
     PE
   CATALOGUED AS
     AMOUNT
     ELEMENT
     PROGRAM
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM PROG_ELE
   DEFAULTED-AS
     CHARACTERS 6
REPORT OF ITEM PROJECT_DESC
   EDITION 1 ENCODED BY MASTER AT 19.23.11 ON 27 NOV 1984
   ALIAS
     PDES
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM PROJECT_DESC
   DEFAULTED-AS
     CHARACTERS 26
REPORT OF ITEM CURR_SCOPE
   EDITION 1 ENCODED BY MASTER AT 19.23.12 ON 27 NOV 1984
   ALIAS
     SCOPE_C
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM CURR_SCOPE
   DEFAULTED-AS
     NUMERIC-CHARACTER 9
REPORT OF ITEM ORI_SCOPE
   EDITION 1 ENCODED BY MASTER AT 19.23.13 ON 27 NOV 1984
   ALIAS
     SCOPE_O
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM ORI_SCOPE
   DEFAULTED-AS
     NUMERIC-CHARACTER 9
REPORT OF ITEM TYPE_FUNDS
   EDITION 1 ENCODED BY MASTER AT 19.23.13 ON 27 NOV 1984
   ALIAS
     TF
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM TYPE_FUNDS
   DEFAULTED-AS
     CHARACTERS 1
REPORT OF ITEM MISSION
   EDITION 1 ENCODED BY MASTER AT 19.23.14 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MISSION
   DEFAULTED-AS
     CHARACTERS 1
REPORT OF ITEM TEMP_PN
   EDITION 1 ENCODED BY MASTER AT 19.23.15 ON 27 NOV 1984
   ALIAS
     TPN
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM TEMP_PN
   DEFAULTED-AS
     CHARACTERS 7
REPORT OF ITEM PERM_PN
   EDITION 1 ENCODED BY MASTER AT 19 23 15 ON 27 NOV 1984
   ALIAS
     PPN
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
```

```
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM PERM_PN
   DEFAULTED-AS
     CHARACTERS 7
REPORT OF ITEM MAIN_FILLER
   EDITION 1 ENCODED BY MASTER AT 19.23.16 ON 27 NOV 1984
   ALIAS
     FILL1
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MAIN_FILLER
   DEFAULTED-AS
     CHARACTERS 3
REPORT OF ITEM RKEY
   EDITION 1 ENCODED BY MASTER AT 19.23.17 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM RKEY
   DEFAULTED-AS
     CHARACTERS 8
REPORT OF ITEM MOB_GROUP
   EDITION 1 ENCODED BY MASTER AT 19.23.18 ON 27 NOV 1984
   ALIAS
     MOB_GP
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MOB_GROUP
   DEFAULTED-AS
     CHARACTERS 1
REPORT OF FILE ZCPPFILE
   EDITION 1 ENCODED BY MASTER AT 19.24.01 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 52 DIRECT REFERENCES
   FILE ZCPPFILE
   PARENT MAIN
   SEGTYPE U
   FORM DEFAULTED-AS
   CONTAINS
      ITEM  PROGRAM_YEAR
      ITEM  AUTH_YR
      ITEM  AUS
      ITEM  CAPY
      ITEM  APS
      ITEM  PEYR
      ITEM  BUDGET_ACT
      ITEM  MPRO_RCD_DT
      ITEM  MPRO_APD_BY
      ITEM  MPRO_APD_DT
      ITEM  SUB
      ITEM  UM
      ITEM  REMARK_2
      ITEM  REMARK_4
      ITEM  PROG_AMT
      ITEM  DATEPA
      ITEM  AUTH_AMT
      ITEM  APPROP_AMT
      ITEM  PUB_LAW_CD
      ITEM  CON_DIR_AMT
      ITEM  CON_STD
      ITEM  REPLACE_CODE
      ITEM  2807C
      ITEM  2807O
      ITEM  2807RQ
      ITEM  DES_DIR_AMT
      ITEM  DES_DISTR_CD
      ITEM  EXEC_STATUS
      ITEM  CMD_PN
      ITEM  AR526_CODE
      ITEM  SORT1
      ITEM  SORT2
      ITEM  ZCP_X1
      ITEM  ZCP_X2
      ITEM  ZCP_X3
      ITEM  ZCP_X4
      ITEM  FN1
      ITEM  FN2
      ITEM  FN3
      ITEM  SPN
      ITEM  PAGE1391
      ITEM  INDXPAGE
      ITEM  SFT
```

```
    ITEM  REMARKA
    ITEM  REMARKB
    ITEM  REMARKC
    ITEM  PCA
    ITEM  CACTION
    ITEM  DRCN
    ITEM  DRCN2
    ITEM  CHGSW
    ITEM  ZCPP_FILLER
REPORT OF ITEM PROGRAM_YEAR
  EDITION 1 ENCODED BY MASTER AT 19.23.21 ON 27 NOV 1984
  ALIAS
    APYR
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM PROGRAM_YEAR
  DEFAULTED-AS
    CHARACTERS 2
REPORT OF ITEM AUTH_YR
  EDITION 1 ENCODED BY MASTER AT 19.23.22 ON 27 NOV 1984
  ALIAS
    AUYR
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM AUTH_YR
  DEFAULTED-AS
    CHARACTERS 2
REPORT OF ITEM AUS
  EDITION 1 ENCODED BY MASTER AT 19.23.23 ON 27 NOV 1984
  ALIAS
    AUTH_SUP
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM AUS
  DEFAULTED-AS
    CHARACTERS 2
REPORT OF ITEM CAPY
  EDITION 1 ENCODED BY MASTER AT 19 23.24 ON 27 NOV 1984
  ALIAS
    CONG_APRV_YR
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM CAPY
  DEFAULTED-AS
    CHARACTERS 2
REPORT OF ITEM APS
  EDITION 1 ENCODED BY MASTER AT 19.23.24 ON 27 NOV 1984
  ALIAS
    APPROP_SUP
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM APS
  DEFAULTED-AS
    CHARACTERS 2
REPORT OF ITEM PEYR
  EDITION 1 ENCODED BY MASTER AT 19 23.25 ON 27 NOV 1984
  ALIAS
    PROG_EXEC_YR
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM PEYR
  DEFAULTED-AS
    CHARACTERS 2
REPORT OF ITEM BUDGET_ACT
  EDITION 1 ENCODED BY MASTER AT 19 23 26 ON 27 NOV 1984
  ALIAS
    BA
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM BUDGET_ACT
  DEFAULTED-AS
    CHARACTERS 13
REPORT OF ITEM MPRO_RCD_DT
  EDITION 1 ENCODED BY MASTER AT 19 23 27 ON 27 NOV 1984
  ALIAS
    MPRO_DT
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM MPRO_RCD_DT
  DEFAULTED-AS
    NUMERIC-CHARACTER 6
```

```
REPORT OF ITEM MPRO_APD_BY
  EDITION 1 ENCODED BY MASTER AT 19.23.27 ON 27 NOV 1984
  ALIAS
    MPRO_AB
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM MPRO_APD_BY
  DEFAULTED-AS
    CHARACTERS 1
REPORT OF ITEM MPRO_APD_DT
  EDITION 1 ENCODED BY MASTER AT 19.23.28 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM MPRO_APD_DT
  DEFAULTED-AS
    NUMERIC-CHARACTER 6
REPORT OF ITEM SUB
  EDITION 1 ENCODED BY MASTER AT 19.23.29 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM SUB
  DEFAULTED-AS
    CHARACTERS 1
REPORT OF ITEM UM
  EDITION 1 ENCODED BY MASTER AT 19.23.30 ON 27 NOV 1984
  ALIAS
    UNIT_OF_MEA
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM UM
  DEFAULTED-AS
    CHARACTERS 2
REPORT OF ITEM REMARK_2
  EDITION 1 ENCODED BY MASTER AT 19.23.31 ON 27 NOV 1984
  ALIAS
    ZCPP_REM2
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM REMARK_2
  DEFAULTED-AS
    CHARACTERS 60
REPORT OF ITEM REMARK_4
  EDITION 1 ENCODED BY MASTER AT 19.23.31 ON 27 NOV 1984
  ALIAS
    ZCPP_REM4
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM REMARK_4
  DEFAULTED-AS
    CHARACTERS 30
REPORT OF ITEM PROG_AMT
  EDITION 2 ENCODED BY MASTER AT 14.10.24 ON 05 DEC 1984
  ALIAS
    PA
  CATALOGUED AS
    AMOUNT
    PROGRAM
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM PROG_AMT
  DEFAULTED-AS
    NUMERIC-CHARACTER 8
REPORT OF ITEM DATEPA
  EDITION 1 ENCODED BY MASTER AT 19.23.33 ON 27 NOV 1984
  ALIAS
    DT_PA_CHG
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM DATEPA
  DEFAULTED-AS
    NUMERIC-CHARACTER 6
REPORT OF ITEM AUTH_AMT
  EDITION 2 ENCODED BY MASTER AT 14.09.09 ON 05 DEC 1984
  ALIAS
    AA
  CATALOGUED AS
    AMOUNT
    AUTHORIZED
    AUTHORITY
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
```

```
    ITEM AUTH_AMT
    DEFAULTED-AS
      NUMERIC-CHARACTER 8
REPORT OF ITEM APPROP_AMT
    EDITION 1 ENCODED BY MASTER AT 19.23.35 ON 27 NOV 1984
    ALIAS
      APPA
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM APPROP_AMT
    DEFAULTED-AS
      NUMERIC-CHARACTER 8
REPORT OF ITEM PUB_LAW_CD
    EDITION 1 ENCODED BY MASTER AT 19.23.35 ON 27 NOV 1984
    ALIAS
      PLCD
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM PUB_LAW_CD
    DEFAULTED-AS
      CHARACTERS 4
REPORT OF ITEM CON_DIR_AMT
    EDITION 1 ENCODED BY MASTER AT 19.23.36 ON 27 NOV 1984
    ALIAS
      CON_AMT
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM CON_DIR_AMT
    DEFAULTED-AS
      PACKED-DECIMAL 13.2
REPORT OF ITEM CON_STD
    EDITION 1 ENCODED BY MASTER AT 19.23.37 ON 27 NOV 1984
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM CON_STD
    DEFAULTED-AS
      CHARACTERS 1
REPORT OF ITEM REPLACE_CODE
    EDITION 1 ENCODED BY MASTER AT 19 23 38 ON 27 NOV 1984
    ALIAS
      RC
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM REPLACE_CODE
    DEFAULTED-AS
      CHARACTERS 1
REPORT OF ITEM 2807C
    EDITION 1 ENCODED BY MASTER AT 19.23.39 ON 27 NOV 1984
    ALIAS
      2807_CONG_DT
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM 2807C
    DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM 2807O
    EDITION 1 ENCODED BY MASTER AT 19 23 39 ON 27 NOV 1984
    ALIAS
      2807_OSD_DT
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM 2807O
    DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM 2807RQ
    EDITION 1 ENCODED BY MASTER AT 19.23.40 ON 27 NOV 1984
    ALIAS
      2807_RQ
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM 2807RQ
    DEFAULTED-AS
      CHARACTERS 1
REPORT OF ITEM DES_DIR_AMT
    EDITION 1 ENCODED BY MASTER AT 19.23.41 ON 27 NOV 1984
    ALIAS
      DES_AMT
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM DES_DIR_AMT
    CONTAINS NO ENTRIES
```

```
REPORT OF ITEM DES_DISTR_CD
  EDITION 1 ENCODED BY MASTER AT 19.23.42 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM DES_DISTR_CD
  DEFAULTED-AS
    CHARACTERS 2
REPORT OF ITEM EXEC_STATUS
  EDITION 1 ENCODED BY MASTER AT 19.23.42 ON 27 NOV 1984
  ALIAS
    EXSTS
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM EXEC_STATUS
  DEFAULTED-AS
    CHARACTERS 1
REPORT OF ITEM CMD_PN
  EDITION 1 ENCODED BY MASTER AT 19.23.43 ON 27 NOV 1984
  ALIAS
    CPN
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM CMD_PN
  DEFAULTED-AS
    CHARACTERS 15
REPORT OF ITEM AR525_CODE
  EDITION 1 ENCODED BY MASTER AT 19.23.44 ON 27 NOV 1984
  ALIAS
    CMTY_CD
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM AR525_CODE
  DEFAULTED-AS
    CHARACTERS 6
REPORT OF ITEM SORT1
  EDITION 1 ENCODED BY MASTER AT 19.23.45 ON 27 NOV 1984
  ALIAS
    S1
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM SORT1
  DEFAULTED-AS
    CHARACTERS 3
REPORT OF ITEM SORT2
  EDITION 1 ENCODED BY MASTER AT 19.23.45 ON 27 NOV 1984
  ALIAS
    S2
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM SORT2
  DEFAULTED-AS
    CHARACTERS 3
REPORT OF ITEM ZCP_X1
  EDITION 1 ENCODED BY MASTER AT 19.23.46 ON 27 NOV 1984
  ALIAS
    ZX1
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM ZCP_X1
  DEFAULTED-AS
    CHARACTERS 1
REPORT OF ITEM ZCP_X2
  EDITION 1 ENCODED BY MASTER AT 19.23.47 ON 27 NOV 1984
  ALIAS
    ZX2
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM ZCP_X2
  DEFAULTED-AS
    CHARACTERS 1
REPORT OF ITEM ZCP_X3
  EDITION 1 ENCODED BY MASTER AT 19.23.48 ON 27 NOV 1984
  ALIAS
    ZX3
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM ZCP_X3
  DEFAULTED-AS
    CHARACTERS 2
REPORT OF ITEM ZCP_X4
  EDITION 1 ENCODED BY MASTER AT 19.23.49 ON 27 NOV 1984
```

```
   ALIAS
     ZX4
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM ZCP_X4
   DEFAULTED-AS
     CHARACTERS 3
REPORT OF ITEM FN1
   EDITION 1 ENCODED BY MASTER AT 19.23.49 ON 27 NOV 1984
   ALIAS
     FOOTNOTE1
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM FN1
   DEFAULTED-AS
     CHARACTERS 1
REPORT OF ITEM FN2
   EDITION 1 ENCODED BY MASTER AT 19.23.50 ON 27 NOV 1984
   ALIAS
     EA
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM FN2
   DEFAULTED-AS
     CHARACTERS 1
REPORT OF ITEM FN3
   EDITION 1 ENCODED BY MASTER AT 19.23.51 ON 27 NOV 1984
   ALIAS
     FOOTNOTE3
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM FN3
   DEFAULTED-AS
     CHARACTERS 1
REPORT OF ITEM SPN
   EDITION 1 ENCODED BY MASTER AT 19.23.52 ON 27 NOV 1984
   ALIAS
     SHORTPN
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM SPN
   DEFAULTED-AS
     CHARACTERS 8
REPORT OF ITEM PAGE1391
   EDITION 1 ENCODED BY MASTER AT 19.23.52 ON 27 NOV 1984
   ALIAS
     GRBK_1391PG
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM PAGE1391
   DEFAULTED-AS
     CHARACTERS 4
REPORT OF ITEM INDXPAGE
   EDITION 1 ENCODED BY MASTER AT 19.23.53 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM INDXPAGE
   DEFAULTED-AS
     CHARACTERS 4
REPORT OF ITEM SFT
   EDITION 1 ENCODED BY MASTER AT 19.23.54 ON 27 NOV 1984
   ALIAS
     SUBFUNDTYPE
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM SFT
   DEFAULTED-AS
     CHARACTERS 2
REPORT OF ITEM REMARKA
   EDITION 1 ENCODED BY MASTER AT 19.23.55 ON 27 NOV 1984
   ALIAS
     RMKA
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM REMARKA
   DEFAULTED-AS
     CHARACTERS 10
REPORT OF ITEM REMARKB
   EDITION 1 ENCODED BY MASTER AT 19.23.55 ON 27 NOV 1984
   ALIAS
     RMKB
```

```
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM REMARKB
   DEFAULTED-AS
     CHARACTERS 10
REPORT OF ITEM REMARKC
   EDITION 1 ENCODED BY MASTER AT 19.23.56 ON 27 NOV 1984
   ALIAS
     RMKC
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM REMARKC
   DEFAULTED-AS
     CHARACTERS 10
REPORT OF ITEM PCA
   EDITION 1 ENCODED BY MASTER AT 19.23.57 ON 27 NOV 1984
   ALIAS
     PREVCONGAUTH
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM PCA
   DEFAULTED-AS
     CHARACTERS 2
REPORT OF ITEM CACTION
   EDITION 1 ENCODED BY MASTER AT 19.23.58 ON 27 NOV 1984
   ALIAS
     CONGACTION
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM CACTION
   DEFAULTED-AS
     CHARACTERS 60
REPORT OF ITEM DRCN
   EDITION 1 ENCODED BY MASTER AT 19.23.58 ON 27 NOV 1984
   ALIAS
     DES_CN_NO1
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM DRCN
   DEFAULTED-AS
     CHARACTERS 3
REPORT OF ITEM DRCN2
   EDITION 1 ENCODED BY MASTER AT 19.23.59 ON 27 NOV 1984
   ALIAS
     DES_CN_NO2
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM DRCN2
   DEFAULTED-AS
     CHARACTERS 3
REPORT OF ITEM CHGSW
   EDITION 1 ENCODED BY MASTER AT 19.24.00 ON 27 NOV 1984
   ALIAS
     CHANGE_SW
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM CHGSW
   DEFAULTED-AS
     CHARACTERS 6
REPORT OF ITEM ZCPP_FILLER
   EDITION 1 ENCODED BY MASTER AT 19.24.01 ON 27 NOV 1984
   ALIAS
     FILL2
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM ZCPP_FILLER
   DEFAULTED-AS
     CHARACTERS 1
REPORT OF FILE ZCP2FILE
   EDITION 1 ENCODED BY MASTER AT 19.24.18 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 15 DIRECT REFERENCES
   FILE ZCP2FILE
   PARENT MAIN
   SEGTYPE U
   FORM DEFAULTED-AS
   CONTAINS
     ITEM  ZRD
     ITEM  CURR1391_DT
     ITEM  1391_DT_DIST
     ITEM  1391_OK
```

```
      ITEM  FORMNO
      ITEM  PROCPA
      ITEM  CONTROL_CODE
      ITEM  PROCFY
      ITEM  COMPONENT
      ITEM  ZCP2_FILLER
      ITEM  MOBPRI
      ITEM  MOB_DIST
      ITEM  LSD
      ITEM  ROD
      ITEM  FILL11
REPORT OF ITEM ZRD
  EDITION 1 ENCODED BY MASTER AT 19.24.07 ON 27 NOV 1984
  ALIAS
    ZCP2_REF_DT
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM ZRD
  DEFAULTED-AS
    NUMERIC-CHARACTER 6
REPORT OF ITEM CURR1391_DT
  EDITION 1 ENCODED BY MASTER AT 19.24.07 ON 27 NOV 1984
  ALIAS
    1391_DT
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM CURR1391_DT
  DEFAULTED-AS
    NUMERIC-CHARACTER 6
REPORT OF ITEM 1391_DT_DIST
  EDITION 1 ENCODED BY MASTER AT 19.24.08 ON 27 NOV 1984
  ALIAS
    1391_DIST
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM 1391_DT_DIST
  DEFAULTED-AS
    NUMERIC-CHARACTER 6
REPORT OF ITEM 1391_OK
  EDITION 1 ENCODED BY MASTER AT 19.24.09 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM 1391_OK
  DEFAULTED-AS
    CHARACTERS 1
REPORT OF ITEM FORMNO
  EDITION 1 ENCODED BY MASTER AT 19.24.09 ON 27 NOV 1984
  ALIAS
    FNO
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM FORMNO
  DEFAULTED-AS
    CHARACTERS 7
REPORT OF ITEM PROCPA
  EDITION 1 ENCODED BY MASTER AT 19.24.10 ON 27 NOV 1984
  ALIAS
    PROCESS_COST
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM PROCPA
  DEFAULTED-AS
    NUMERIC-CHARACTER 8
REPORT OF ITEM CONTROL_CODE
  EDITION 1 ENCODED BY MASTER AT 19.24.11 ON 27 NOV 1984
  ALIAS
    CNTRC
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM CONTROL_CODE
  DEFAULTED-AS
    CHARACTERS 1
REPORT OF ITEM PROCFY
  EDITION 1 ENCODED BY MASTER AT 19.24.12 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM PROCFY
  DEFAULTED-AS
    CHARACTERS 4
REPORT OF ITEM COMPONENT
  EDITION 1 ENCODED BY MASTER AT 19.24.12 ON 27 NOV 1984
```

```
   ALIAS
      COMPT
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM COMPONENT
   DEFAULTED-AS
      CHARACTERS 4
REPORT OF ITEM ZCP2_FILLER
   EDITION 1 ENCODED BY MASTER AT 19.24.13 ON 27 NOV 1984
   ALIAS
      FILL3
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM ZCP2_FILLER
   DEFAULTED-AS
      CHARACTERS 3
REPORT OF ITEM MOBPRI
   EDITION 1 ENCODED BY MASTER AT 19.24.14 ON 27 NOV 1984
   ALIAS
      DA_PRI
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MOBPRI
   DEFAULTED-AS
      CHARACTERS 5
REPORT OF ITEM MOB_DIST
   EDITION 1 ENCODED BY MASTER AT 19.24.14 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MOB_DIST
   DEFAULTED-AS
      CHARACTERS 2
REPORT OF ITEM LSD
   EDITION 1 ENCODED BY MASTER AT 19.24.15 ON 27 NOV 1984
   ALIAS
      LATE_ST_DT
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM LSD
   DEFAULTED-AS
      CHARACTERS 5
REPORT OF ITEM ROD
   EDITION 1 ENCODED BY MASTER AT 19.24.16 ON 27 NOV 1984
   ALIAS
      RQD_OCP_DT
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM ROD
   DEFAULTED-AS
      CHARACTERS 5
REPORT OF ITEM FILL11
   EDITION 1 ENCODED BY MASTER AT 19.24.17 ON 27 NOV 1984
   ALIAS
      MOB_FILLER
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM FILL11
   DEFAULTED-AS
      CHARACTERS 2
REPORT OF FILE ZCP3FILE
   EDITION 1 ENCODED BY MASTER AT 19.24.33 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 18 DIRECT REFERENCES
   FILE ZCP3FILE
   PARENT MAIN
   SEGTYPE U
   FORM DEFAULTED-AS
   CONTAINS
      ITEM  LONG_DESC
      ITEM  AUTH_REQ
      ITEM  APPR_REQ
      ITEM  AUTH_CODE
      ITEM  APPR_CODE
      ITEM  OSD_NOTE
      ITEM  ARMY_REQ
      ITEM  SASCAUTH
      ITEM  HASCAUTH
      ITEM  SAC_APPR
      ITEM  HAC_APPR
      ITEM  CONGAUTH
      ITEM  CONFAPPR
```

```
      ITEM   AMT1
      ITEM   AMT2
      ITEM   AMT3
      ITEM   AMT4
      ITEM   ZCP3_FILLER
REPORT OF ITEM LONG_DESC
  EDITION 1 ENCODED BY MASTER AT 19.24.20 ON 27 NOV 1984
  ALIAS
    LDESC
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM LONG_DESC
  DEFAULTED-AS
    CHARACTERS 42
REPORT OF ITEM AUTH_REQ
  EDITION 1 ENCODED BY MASTER AT 19.24.21 ON 27 NOV 1984
  ALIAS
    REQAUTH
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM AUTH_REQ
  DEFAULTED-AS
    NUMERIC-CHARACTER 8
REPORT OF ITEM APPR_REQ
  EDITION 1 ENCODED BY MASTER AT 19.24.22 ON 27 NOV 1984
  ALIAS
    REQAPPR
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM APPR_REQ
  DEFAULTED-AS
    NUMERIC-CHARACTER 8
REPORT OF ITEM AUTH_CODE
  EDITION 1 ENCODED BY MASTER AT 19.24.22 ON 27 NOV 1984
  ALIAS
    AUTHCD
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM AUTH_CODE
  DEFAULTED-AS
    CHARACTERS 1
REPORT OF ITEM APPR_CODE
  EDITION 1 ENCODED BY MASTER AT 19.24.23 ON 27 NOV 1984
  ALIAS
    APPRCD
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM APPR_CODE
  DEFAULTED-AS
    CHARACTERS 1
REPORT OF ITEM OSD_NOTE
  EDITION 1 ENCODED BY MASTER AT 19.24.24 ON 27 NOV 1984
  ALIAS
    OSDN
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM OSD_NOTE
  DEFAULTED-AS
    CHARACTERS 25
REPORT OF ITEM ARMY_REQ
  EDITION 1 ENCODED BY MASTER AT 19.24.24 ON 27 NOV 1984
  ALIAS
    ARMQ
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM ARMY_REQ
  DEFAULTED-AS
    NUMERIC-CHARACTER 8
REPORT OF ITEM SASCAUTH
  EDITION 1 ENCODED BY MASTER AT 19.24.25 ON 27 NOV 1984
  ALIAS
    SAUTH
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM SASCAUTH
  DEFAULTED-AS
    NUMERIC-CHARACTER 8
REPORT OF ITEM HASCAUTH
  EDITION 1 ENCODED BY MASTER AT 19 24 26 ON 27 NOV 1984
  ALIAS
    HAUTH
```

```
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HASCAUTH
   DEFAULTED-AS
     NUMERIC-CHARACTER 8
REPORT OF ITEM SAC_APPR
   EDITION 1 ENCODED BY MASTER AT 19.24.27 ON 27 NOV 1984
   ALIAS
     SAPPR
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM SAC_APPR
   DEFAULTED-AS
     NUMERIC-CHARACTER 8
REPORT OF ITEM HAC_APPR
   EDITION 1 ENCODED BY MASTER AT 19.24.27 ON 27 NOV 1984
   ALIAS
     HAPPR
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HAC_APPR
   DEFAULTED-AS
     NUMERIC-CHARACTER 8
REPORT OF ITEM CONGAUTH
   EDITION 1 ENCODED BY MASTER AT 19.24.28 ON 27 NOV 1984
   ALIAS
     CAUTH
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM CONGAUTH
   DEFAULTED-AS
     NUMERIC-CHARACTER 8
REPORT OF ITEM CONFAPPR
   EDITION 1 ENCODED BY MASTER AT 19.24.29 ON 27 NOV 1984
   ALIAS
     CAPPR
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM CONFAPPR
   DEFAULTED-AS
     NUMERIC-CHARACTER 8
REPORT OF ITEM AMT1
   EDITION 1 ENCODED BY MASTER AT 19.24.30 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM AMT1
   DEFAULTED-AS
     NUMERIC-CHARACTER 8
REPORT OF ITEM AMT2
   EDITION 1 ENCODED BY MASTER AT 19.24.30 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM AMT2
   DEFAULTED-AS
     NUMERIC-CHARACTER 8
REPORT OF ITEM AMT3
   EDITION 1 ENCODED BY MASTER AT 19.24.31 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM AMT3
   DEFAULTED-AS
     NUMERIC-CHARACTER 8
REPORT OF ITEM AMT4
   EDITION 1 ENCODED BY MASTER AT 19.24.32 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM AMT4
   DEFAULTED-AS
     NUMERIC-CHARACTER 8
REPORT OF ITEM ZCP3_FILLER
   EDITION 1 ENCODED BY MASTER AT 19.24.32 ON 27 NOV 1984
   ALIAS
     FILL4
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM ZCP3_FILLER
   DEFAULTED-AS
     CHARACTERS 3
REPORT OF FILE AMPERS1
   EDITION 1 ENCODED BY MASTER AT 19.24.41 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
```

```
   THIS MEMBER CONTAINS 6 DIRECT REFERENCES
   FILE AMPERS1
   PARENT MAIN
   SEGTYPE U
   FORM DEFAULTED-AS
   CONTAINS
      ITEM  DES_PERCENT
      ITEM  CWE_AMT
      ITEM  CONCOM_DT
      ITEM  DES_COMP_DT
      ITEM  DES_ST_DT
      ITEM  AMPER_FILLER
REPORT OF ITEM DES_PERCENT
   EDITION 2 ENCODED BY MASTER AT 14.13.09 ON 06 DEC 1984
   ALIAS
      DES_%
   CATALOGUED AS
      PROGRESS
      DESIGN
      COMPLETION
      PERCENT
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM DES_PERCENT
   DEFAULTED-AS
      CHARACTERS 3
REPORT OF ITEM CWE_AMT
   EDITION 2 ENCODED BY MASTER AT 14.12.25 ON 06 DEC 1984
   ALIAS
      CWE
   CATALOGUED AS
      CURRENT
      ESTIMATE
      AMOUNT
      PROJECT
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM CWE_AMT
   DEFAULTED-AS
      NUMERIC-CHARACTER 8
REPORT OF ITEM CONCOM_DT
   EDITION 1 ENCODED BY MASTER AT 19.24.36 ON 27 NOV 1984
   ALIAS
      CONCEPT
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM CONCOM_DT
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM DES_COMP_DT
   EDITION 2 ENCODED BY MASTER AT 14.14.59 ON 06 DEC 1984
   CATALOGUED AS
      DESIGN
      COMPLETION
      DATE
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM DES_COMP_DT
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM DES_ST_DT
   EDITION 2 ENCODED BY MASTER AT 14.13.46 ON 06 DEC 1984
   ALIAS
      DES_SD
   CATALOGUED AS
      DESIGN
      START
      DATE
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM DES_ST_DT
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM AMPER_FILLER
   EDITION 1 ENCODED BY MASTER AT 19.24.40 ON 27 NOV 1984
   ALIAS
      FILL6
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM AMPER_FILLER
   DEFAULTED-AS
```

```
      CHARACTERS 1
REPORT OF FILE ACEFILE
   EDITION 1 ENCODED BY MASTER AT 19.25.06 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 26 DIRECT REFERENCES
   FILE ACEFILE
   PARENT MAIN
   SEGTYPE U
   FORM DEFAULTED-AS
   CONTAINS
      ITEM  CRRC
      ITEM  LEVEL
      ITEM  ACE_PA
      ITEM  PA_DATE
      ITEM  PROP
      ITEM  REMARK_1
      ITEM  ACEWORK1
      ITEM  ACEWORK2
      ITEM  ACEWORK3
      ITEM  ACEPRB_PRI
      ITEM  ZB
      ITEM  CRRC_DATE
      ITEM  ACE_PY
      ITEM  PY_DATE
      ITEM  ACE_CMD_PRI
      ITEM  ACE_D_REL
      ITEM  ACE_DCD
      ITEM  ZDTCD1
      ITEM  ZDTCD2
      ITEM  ZDTCD4
      ITEM  ZDTCD5
      ITEM  ZDTCD6
      ITEM  ZDTCD8
      ITEM  ZDTCD9
      ITEM  PROP_PRIOR
      ITEM  CRRC_PRIOR
REPORT OF ITEM CRRC
   EDITION 1 ENCODED BY MASTER AT 19.24.43 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM CRRC
   DEFAULTED-AS
      CHARACTERS 1
REPORT OF ITEM LEVEL
   EDITION 1 ENCODED BY MASTER AT 19.24.44 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM LEVEL
   DEFAULTED-AS
      CHARACTERS 1
REPORT OF ITEM ACE_PA
   EDITION 1 ENCODED BY MASTER AT 19.24.44 ON 27 NOV 1984
   ALIAS
      APROJA
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM ACE_PA
   DEFAULTED-AS
      NUMERIC-CHARACTER 8
REPORT OF ITEM PA_DATE
   EDITION 1 ENCODED BY MASTER AT 19.24.45 ON 27 NOV 1984
   ALIAS
      DATE_PA_CHGE
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM PA_DATE
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM PROP
   EDITION 1 ENCODED BY MASTER AT 19.24.46 ON 27 NOV 1984
   ALIAS
      PROPONENT
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM PROP
   DEFAULTED-AS
      CHARACTERS 3
REPORT OF ITEM REMARK_1
   EDITION 1 ENCODED BY MASTER AT 19.24.47 ON 27 NOV 1984
   ALIAS
      ACE_REM
```

```
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM REMARK_1
    DEFAULTED-AS
      CHARACTERS 60
REPORT OF ITEM ACEWORK1
    EDITION 1 ENCODED BY MASTER AT 19.24.48 ON 27 NOV 1984
    ALIAS
      AWK1
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM ACEWORK1
    DEFAULTED-AS
      CHARACTERS 4
REPORT OF ITEM ACEWORK2
    EDITION 1 ENCODED BY MASTER AT 19.24.49 ON 27 NOV 1984
    ALIAS
      AWK2
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM ACEWORK2
    DEFAULTED-AS
      CHARACTERS 6
REPORT OF ITEM ACEWORK3
    EDITION 1 ENCODED BY MASTER AT 19.24.50 ON 27 NOV 1984
    ALIAS
      AWK3
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM ACEWORK3
    DEFAULTED-AS
      CHARACTERS 3
REPORT OF ITEM ACEPRB_PRI
    EDITION 1 ENCODED BY MASTER AT 19.24.51 ON 27 NOV 1984
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM ACEPRB_PRI
    DEFAULTED-AS
      CHARACTERS 4
REPORT OF ITEM ZB
    EDITION 1 ENCODED BY MASTER AT 19.24.51 ON 27 NOV 1984
    ALIAS
      ZERO_BUDG
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM ZB
    DEFAULTED-AS
      CHARACTERS 2
REPORT OF ITEM CRRC_DATE
    EDITION 1 ENCODED BY MASTER AT 19.24.52 ON 27 NOV 1984
    ALIAS
      CRD
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM CRRC_DATE
    DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM ACE_PY
    EDITION 1 ENCODED BY MASTER AT 19.24.54 ON 27 NOV 1984
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM ACE_PY
    DEFAULTED-AS
      CHARACTERS 2
REPORT OF ITEM PY_DATE
    EDITION 1 ENCODED BY MASTER AT 19.24.55 ON 27 NOV 1984
    ALIAS
      DATE_PY_CHG
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM PY_DATE
    DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM ACE_CMD_PRI
    EDITION 1 ENCODED BY MASTER AT 19.24.56 ON 27 NOV 1984
    ALIAS
      ACMDP
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM ACE_CMD_PRI
    DEFAULTED-AS
```

```
      CHARACTERS 5
REPORT OF ITEM ACE_D_REL
   EDITION 1 ENCODED BY MASTER AT 19.24.57 ON 27 NOV 1984
   ALIAS
      ACEDR
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM ACE_D_REL
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM ACE_DCD
   EDITION 1 ENCODED BY MASTER AT 19.24.57 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM ACE_DCD
   DEFAULTED-AS
      CHARACTERS 1
REPORT OF ITEM ZDTCD1
   EDITION 1 ENCODED BY MASTER AT 19.24.58 ON 27 NOV 1984
   ALIAS
      ZCD1
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM ZDTCD1
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM ZDTCD2
   EDITION 1 ENCODED BY MASTER AT 19.24.59 ON 27 NOV 1984
   ALIAS
      ZCD2
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM ZDTCD2
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM ZDTCD4
   EDITION 1 ENCODED BY MASTER AT 19.25.00 ON 27 NOV 1984
   ALIAS
      ZCD4
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM ZDTCD4
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM ZDTCD5
   EDITION 1 ENCODED BY MASTER AT 19.25.01 ON 27 NOV 1984
   ALIAS
      ZCD5
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM ZDTCD5
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM ZDTCD6
   EDITION 1 ENCODED BY MASTER AT 19.25.02 ON 27 NOV 1984
   ALIAS
      ZCD6
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM ZDTCD6
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM ZDTCD8
   EDITION 1 ENCODED BY MASTER AT 19.25.02 ON 27 NOV 1984
   ALIAS
      ZCD8
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM ZDTCD8
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM ZDTCD9
   EDITION 1 ENCODED BY MASTER AT 19.25.03 ON 27 NOV 1984
   ALIAS
      ZCD9
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM ZDTCD9
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM PROP_PRIOR
```

```
   EDITION 1 ENCODED BY MASTER AT 10.25.04 ON 27 NOV 1984
   ALIAS
     P_PRI
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM PROP_PRIOR
   DEFAULTED-AS
     CHARACTERS 4
REPORT OF ITEM CRRC_PRIOR
   EDITION 1 ENCODED BY MASTER AT 10.25.05 ON 27 NOV 1984
   ALIAS
     CRRC_PRI
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM CRRC_PRIOR
   DEFAULTED-AS
     CHARACTERS 4
REPORT OF FILE MPCAFILE
   EDITION 1 ENCODED BY MASTER AT 10.25.31 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 25 DIRECT REFERENCES
   FILE MPCAFILE
   PARENT MAIN
   SEGTYPE U
   FORM DEFAULTED-AS
   CONTAINS
      ITEM  EX_AWARD_DT
      ITEM  EXEC_CODE
      ITEM  MPCA_REM
      ITEM  REMARK_6A
      ITEM  REMARK_6
      ITEM  CDTCD1
      ITEM  CDTCD2
      ITEM  CDTCD4
      ITEM  CDTCD5
      ITEM  CDTCD6
      ITEM  CDTCD8
      ITEM  CDTCD9
      ITEM  BIDOD
      ITEM  NOBID
      ITEM  GOVT_EST
      ITEM  LBID
      ITEM  HBID
      ITEM  DIRCD
      ITEM  HL_DES
      ITEM  MPC_DCD
      ITEM  CONS_COMP_DT
      ITEM  BOD
      ITEM  DES_DIR_DT
      ITEM  FORECAST
      ITEM  AWD_CWE
REPORT OF ITEM EX_AWARD_DT
   EDITION 1 ENCODED BY MASTER AT 10.25.09 ON 27 NOV 1984
   ALIAS
     CONST_AW_DT
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM EX_AWARD_DT
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM EXEC_CODE
   EDITION 1 ENCODED BY MASTER AT 10.25.10 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM EXEC_CODE
   DEFAULTED-AS
     CHARACTERS 1
REPORT OF ITEM MPCA_REM
   EDITION 1 ENCODED BY MASTER AT 10.25.11 ON 27 NOV 1984
   ALIAS
     REMARK_6
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MPCA_REM
   DEFAULTED-AS
     CHARACTERS 30
REPORT OF ITEM REMARK_6A
   EDITION 1 ENCODED BY MASTER AT 10.25.12 ON 27 NOV 1984
   ALIAS
     RMK6
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
```

```
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM REMARK_5A
   DEFAULTED-AS
      CHARACTERS 30
REPORT OF ITEM REMARK_6
   EDITION 1 ENCODED BY MASTER AT 19 25 12 ON 27 NOV 1984
   ALIAS
      RMK6
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM REMARK_6
   DEFAULTED-AS
      CHARACTERS 60
REPORT OF ITEM CDTCD1
   EDITION 1 ENCODED BY MASTER AT 19 25 13 ON 27 NOV 1984
   ALIAS
      CD1
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM CDTCD1
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM CDTCD2
   EDITION 1 ENCODED BY MASTER AT 19 25 14 ON 27 NOV 1984
   ALIAS
      CD2
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM CDTCD2
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM CDTCD4
   EDITION 1 ENCODED BY MASTER AT 19 25 15 ON 27 NOV 1984
   ALIAS
      CD4
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM CDTCD4
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM CDTCD5
   EDITION 1 ENCODED BY MASTER AT 19 25 16 ON 27 NOV 1984
   ALIAS
      CD5
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM CDTCD5
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM CDTCD6
   EDITION 1 ENCODED BY MASTER AT 19 25 17 ON 27 NOV 1984
   ALIAS
      CD6
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM CDTCD6
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM CDTCD8
   EDITION 1 ENCODED BY MASTER AT 19 25 18 ON 27 NOV 1984
   ALIAS
      CD8
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM CDTCD8
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM CDTCD9
   EDITION 1 ENCODED BY MASTER AT 19 25 18 ON 27 NOV 1984
   ALIAS
      CD9
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM CDTCD9
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM BIDOD
   EDITION 1 ENCODED BY MASTER AT 19 25 19 ON 27 NOV 1984
   ALIAS
      BID_OPEN_DT
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
```

```
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM BIDOD
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM NOBID
   EDITION 1 ENCODED BY MASTER AT 19.25.20 ON 27 NOV 1984
   ALIAS
     NO_BIDDERS
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM NOBID
   DEFAULTED-AS
     NUMERIC-CHARACTER 4
REPORT OF ITEM GOVT_EST
   EDITION 1 ENCODED BY MASTER AT 19.25.21 ON 27 NOV 1984
   ALIAS
     GEST
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM GOVT_EST
   DEFAULTED-AS
     NUMERIC-CHARACTER 8
REPORT OF ITEM LBID
   EDITION 1 ENCODED BY MASTER AT 19.25.22 ON 27 NOV 1984
   ALIAS
     LOW_BID
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM LBID
   DEFAULTED-AS
     NUMERIC-CHARACTER 8
REPORT OF ITEM HBID
   EDITION 1 ENCODED BY MASTER AT 19.25.23 ON 27 NOV 1984
   ALIAS
     HIGH_BID
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HBID
   DEFAULTED-AS
     NUMERIC-CHARACTER 8
REPORT OF ITEM DIRCD
   EDITION 1 ENCODED BY MASTER AT 19.25.23 ON 27 NOV 1984
   ALIAS
     ZCPP_DCD
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM DIRCD
   DEFAULTED-AS
     CHARACTERS 1
REPORT OF ITEM HL_DES
   EDITION 1 ENCODED BY MASTER AT 19.25.24 ON 27 NOV 1984
   ALIAS
     IH_DES
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HL_DES
   DEFAULTED-AS
     CHARACTERS 1
REPORT OF ITEM MPC_DCD
   EDITION 1 ENCODED BY MASTER AT 19.25.25 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MPC_DCD
   DEFAULTED-AS
     CHARACTERS 1
REPORT OF ITEM CONS_COMP_DT
   EDITION 1 ENCODED BY MASTER AT 19.25.26 ON 27 NOV 1984
   ALIAS
     CNCND
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM CONS_COMP_DT
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM BOD
   EDITION 1 ENCODED BY MASTER AT 19.25.27 ON 27 NOV 1984
   ALIAS
     BEN_OCC_DT
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM BOD
```

```
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM DES_DIR_DT
   EDITION 1 ENCODED BY MASTER AT 19.25.28 ON 27 NOV 1984
   ALIAS
      DES_DR
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM DES_DIR_DT
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM FORECAST
   EDITION 1 ENCODED BY MASTER AT 19.25.29 ON 27 NOV 1984
   ALIAS
      FCST_AWD_DT
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM FORECAST
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM AWD_CWE
   EDITION 1 ENCODED BY MASTER AT 19.25.30 ON 27 NOV 1984
   ALIAS
      AWD_CWE_AMT
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM AWD_CWE
   DEFAULTED-AS
      NUMERIC-CHARACTER 8
REPORT OF FILE MPESFILE
   EDITION 1 ENCODED BY MASTER AT 19.25.47 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 16 DIRECT REFERENCES
   FILE MPESFILE
   PARENT MAIN
   SEGTYPE U
   FORM DEFAULTED-AS
   CONTAINS
      ITEM   EST_COST
      ITEM   SUPER_ADMIN
      ITEM   PER_CONT
      ITEM   EST_DT
      ITEM   PER_SA
      ITEM   PROJ_COST_DT
      ITEM   PROJ_COST
      ITEM   PROJ_COST_CD
      ITEM   SOLAR
      ITEM   PDB_DT
      ITEM   PDB_RDQ
      ITEM   CAT_E_EQ
      ITEM   CONT
      ITEM   PER_DSGN
      ITEM   MPES_COMMENT
      ITEM   MPES_FILLER
REPORT OF ITEM EST_COST
   EDITION 1 ENCODED BY MASTER AT 19.25.34 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM EST_COST
   DEFAULTED-AS
      NUMERIC-CHARACTER 8
REPORT OF ITEM SUPER_ADMIN
   EDITION 1 ENCODED BY MASTER AT 19.25.35 ON 27 NOV 1984
   ALIAS
      SA
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM SUPER_ADMIN
   DEFAULTED-AS
      NUMERIC-CHARACTER 8
REPORT OF ITEM PER_CONT
   EDITION 1 ENCODED BY MASTER AT 19.25.37 ON 27 NOV 1984
   ALIAS
      PCONT
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM PER_CONT
   DEFAULTED-AS
      PACKED-DECIMAL 4.1
REPORT OF ITEM EST_DT
   EDITION 1 ENCODED BY MASTER AT 19.25.37 ON 27 NOV 1984
```

```
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM EST_DT
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM PER_SA
   EDITION 1 ENCODED BY MASTER AT 19.25.38 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM PER_SA
   DEFAULTED-AS
     PACKED-DECIMAL 3.1
REPORT OF ITEM PROJ_COST_DT
   EDITION 1 ENCODED BY MASTER AT 19.25.39 ON 27 NOV 1984
   ALIAS
     PCDT
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM PROJ_COST_DT
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM PROJ_COST
   EDITION 1 ENCODED BY MASTER AT 19.25.40 ON 27 NOV 1984
   ALIAS
     PC
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM PROJ_COST
   DEFAULTED-AS
     NUMERIC-CHARACTER 8
REPORT OF ITEM PROJ_COST_CD
   EDITION 1 ENCODED BY MASTER AT 19.25.40 ON 27 NOV 1984
   ALIAS
     PCCD
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM PROJ_COST_CD
   DEFAULTED-AS
     CHARACTERS 1
REPORT OF ITEM SOLAR
   EDITION 1 ENCODED BY MASTER AT 19.25.41 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM SOLAR
   DEFAULTED-AS
     NUMERIC-CHARACTER 8
REPORT OF ITEM PDB_DT
   EDITION 1 ENCODED BY MASTER AT 19.25.42 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM PDB_DT
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM PDB_RDQ
   EDITION 1 ENCODED BY MASTER AT 19.25.43 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM PDB_RDQ
   DEFAULTED-AS
     CHARACTERS 1
REPORT OF ITEM CAT_E_EQ
   EDITION 1 ENCODED BY MASTER AT 19.25.43 ON 27 NOV 1984
   ALIAS
     CAT_E
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM CAT_E_EQ
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM CONT
   EDITION 1 ENCODED BY MASTER AT 19.25.44 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM CONT
   DEFAULTED-AS
     NUMERIC-CHARACTER 8
REPORT OF ITEM PER_DSGN
   EDITION 1 ENCODED BY MASTER AT 19.25.45 ON 27 NOV 1984
   ALIAS
     PDSN
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
```

```
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM PER_DSGN
   DEFAULTED-AS
     PACKED-DECIMAL 5.1
REPORT OF ITEM MPES_COMMENT
   EDITION 1 ENCODED BY MASTER AT 19.25.46 ON 27 NOV 1984
   ALIAS
     MCOM
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MPES_COMMENT
   DEFAULTED-AS
     CHARACTERS 20
REPORT OF ITEM MPES_FILLER
   EDITION 1 ENCODED BY MASTER AT 19.25.47 ON 27 NOV 1984
   ALIAS
     FILL7
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MPES_FILLER
   DEFAULTED-AS
     CHARACTERS 2
REPORT OF FILE MPENFILE
   EDITION 1 ENCODED BY MASTER AT 19.25.57 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 10 DIRECT REFERENCES
   FILE MPENFILE
   PARENT MAIN
   SEGTYPE U
   FORM DEFAULTED-AS
   CONTAINS
      ITEM  REMARK_3
      ITEM  DT_REL_CD1
      ITEM  DT_REL_CD2
      ITEM  DEI_DT
      ITEM  MPEN_IND
      ITEM  DEI_RQ
      ITEM  MPEN_DCD
      ITEM  ENREVDT
      ITEM  ENRN
      ITEM  MPEN_FILLER
REPORT OF ITEM REMARK_3
   EDITION 1 ENCODED BY MASTER AT 19.25.50 ON 27 NOV 1984
   ALIAS
     MPEN_REM
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM REMARK_3
   DEFAULTED-AS
     CHARACTERS 60
REPORT OF ITEM DT_REL_CD1
   EDITION 1 ENCODED BY MASTER AT 19.25.51 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM DT_REL_CD1
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM DT_REL_CD2
   EDITION 1 ENCODED BY MASTER AT 19.25.51 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM DT_REL_CD2
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM DEI_DT
   EDITION 1 ENCODED BY MASTER AT 19.25.52 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM DEI_DT
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM MPEN_IND
   EDITION 1 ENCODED BY MASTER AT 19.25.53 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MPEN_IND
   DEFAULTED-AS
     CHARACTERS 1
REPORT OF ITEM DEI_RQ
   EDITION 1 ENCODED BY MASTER AT 19.25.53 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
```

```
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM DEI_RQ
   DEFAULTED-AS
      CHARACTERS 1
REPORT OF ITEM MPEN_DCD
   EDITION 1 ENCODED BY MASTER AT 19 26 54 ON 27 NOV 1984
   ALIAS
      MPED
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MPEN_DCD
   DEFAULTED-AS
      CHARACTERS 1
REPORT OF ITEM ENREVDT
   EDITION 1 ENCODED BY MASTER AT 19 26 55 ON 27 NOV 1984
   ALIAS
      DT_REL_REV
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM ENREVDT
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM ENRN
   EDITION 1 ENCODED BY MASTER AT 19 26 56 ON 27 NOV 1984
   ALIAS
      ENG_REV_NUM
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM ENRN
   DEFAULTED-AS
      CHARACTERS 2
REPORT OF ITEM MPEN_FILLER
   EDITION 1 ENCODED BY MASTER AT 19 26 56 ON 27 NOV 1984
   ALIAS
      FILL6
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MPEN_FILLER
   DEFAULTED-AS
      CHARACTERS 3
REPORT OF FILE SIGFILE
   EDITION 1 ENCODED BY MASTER AT 19 26 03 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 5 DIRECT REFERENCES
   FILE SIGFILE
   PARENT MAIN
   SEGTYPE U
   FORM DEFAULTED-AS
   CONTAINS
      ITEM  COM_COST
      ITEM  COM_STAT
      ITEM  COM_DATE
      ITEM  TELER_NO
      ITEM  SIG_FILLER
REPORT OF ITEM COM_COST
   EDITION 1 ENCODED BY MASTER AT 19 25 59 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM COM_COST
   DEFAULTED-AS
      NUMERIC-CHARACTER 9
REPORT OF ITEM COM_STAT
   EDITION 1 ENCODED BY MASTER AT 19 26 00 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM COM_STAT
   DEFAULTED-AS
      CHARACTERS 3
REPORT OF ITEM COM_DATE
   EDITION 1 ENCODED BY MASTER AT 19 26 01 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM COM_DATE
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM TELER_NO
   EDITION 1 ENCODED BY MASTER AT 19 26 01 ON 27 NOV 1984
   ALIAS
      TNO
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
```

```
   ITEM TELER_NO
   DEFAULTED-AS
      CHARACTERS 16
REPORT OF ITEM SIG_FILLER
   EDITION 1 ENCODED BY MASTER AT 19.26.02 ON 27 NOV 1984
   ALIAS
      FILL9
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM SIG_FILLER
   DEFAULTED-AS
      CHARACTERS 1
REPORT OF FILE REPRFILE
   EDITION 1 ENCODED BY MASTER AT 19.26.11 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 8 DIRECT REFERENCES
   FILE REPRFILE
   PARENT MAIN
   SEGTYPE S1
   SORT-KEY
      ITEM  ZCP_PA_SPN
   FORM DEFAULTED-AS
   CONTAINS
      ITEM  ZCP_PA_SPN
      ITEM  ZS_SDT
      ITEM  ZS_REM
      ITEM  ZS_PROG_AMT
      ITEM  ZS_AUTH_AMT
      ITEM  ZS_APPR_AMT
      ITEM  REPR_FILLER
REPORT OF ITEM ZCP_PA_SPN
   EDITION 1 ENCODED BY MASTER AT 19.26.05 ON 27 NOV 1984
   ALIAS
      ZSPN
   THIS MEMBER IS DIRECTLY REFERRED TO 2 TIMES
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM ZCP_PA_SPN
   DEFAULTED-AS
      CHARACTERS 3
REPORT OF ITEM ZS_SDT
   EDITION 1 ENCODED BY MASTER AT 19.26.05 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM ZS_SDT
   DEFAULTED-AS
      CHARACTERS 12
REPORT OF ITEM ZS_REM
   EDITION 1 ENCODED BY MASTER AT 19.26.06 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM ZS_REM
   DEFAULTED-AS
      CHARACTERS 26
REPORT OF ITEM ZS_PROG_AMT
   EDITION 1 ENCODED BY MASTER AT 19.26.07 ON 27 NOV 1984
   ALIAS
      ZSPA
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM ZS_PROG_AMT
   DEFAULTED-AS
      NUMERIC-CHARACTER 8
REPORT OF ITEM ZS_AUTH_AMT
   EDITION 1 ENCODED BY MASTER AT 19.26.08 ON 27 NOV 1984
   ALIAS
      ZSAUA
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM ZS_AUTH_AMT
   DEFAULTED-AS
      NUMERIC-CHARACTER 8
REPORT OF ITEM ZS_APPR_AMT
   EDITION 1 ENCODED BY MASTER AT 19.26.08 ON 27 NOV 1984
   ALIAS
      ZSAPA
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM ZS_APPR_AMT
   DEFAULTED-AS
      NUMERIC-CHARACTER 8
REPORT OF ITEM REPR_FILLER
```

```
    EDITION 1 ENCODED BY MASTER AT 19.26.10 ON 27 NOV 1984
    ALIAS
      FILL10
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM REPR_FILLER
    DEFAULTED-AS
      CHARACTERS 3
REPORT OF FILE SUBPFILE
    EDITION 1 ENCODED BY MASTER AT 19.26.31 ON 27 NOV 1984
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 19 DIRECT REFERENCES
    FILE SUBPFILE
    PARENT MAIN
    SEGTYPE S1
    SORT-KEY
      ITEM  AMPRS_SPN
    FORM DEFAULTED-AS
    CONTAINS
      ITEM  AMPRS_SPN
      ITEM  SUBAMT
      ITEM  SUBEXYR
      ITEM  SUB_UNIT_MEA
      ITEM  SUBSCOPE
      ITEM  SUBITEM_DESC
      ITEM  SUB_CONTR
      ITEM  SUB_REM
      ITEM  SCH_AWARD_DT
      ITEM  SUB_AWARD_DT
      ITEM  SUBCAD
      ITEM  SUBBOD
      ITEM  SUBBIDOD
      ITEM  SUBNOBID
      ITEM  SUBGEST
      ITEM  SUBLBID
      ITEM  SUBHBID
      ITEM  SAWD_CWE
REPORT OF ITEM AMPRS_SPN
    EDITION 1 ENCODED BY MASTER AT 19.26.13 ON 27 NOV 1984
    THIS MEMBER IS DIRECTLY REFERRED TO 2 TIMES
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM AMPRS_SPN
    DEFAULTED-AS
      CHARACTERS 3
REPORT OF ITEM SUBAMT
    EDITION 1 ENCODED BY MASTER AT 19.26.14 ON 27 NOV 1984
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM SUBAMT
    DEFAULTED-AS
      NUMERIC-CHARACTER 8
REPORT OF ITEM SUBEXYR
    EDITION 1 ENCODED BY MASTER AT 19.26.15 ON 27 NOV 1984
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM SUBEXYR
    DEFAULTED-AS
      CHARACTERS 4
REPORT OF ITEM SUB_UNIT_MEA
    EDITION 1 ENCODED BY MASTER AT 19.26.16 ON 27 NOV 1984
    ALIAS
      SUB_UM
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM SUB_UNIT_MEA
    DEFAULTED-AS
      CHARACTERS 2
REPORT OF ITEM SUBSCOPE
    EDITION 1 ENCODED BY MASTER AT 19.26.16 ON 27 NOV 1984
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM SUBSCOPE
    DEFAULTED-AS
      NUMERIC-CHARACTER 7
REPORT OF ITEM SUBITEM_DESC
    EDITION 1 ENCODED BY MASTER AT 19.26.17 ON 27 NOV 1984
    ALIAS
      SDESC
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM SUBITEM_DESC
```

```
    DEFAULTED-AS
      CHARACTERS 26
REPORT OF ITEM SUB_CONTR
    EDITION 1 ENCODED BY MASTER AT 19.26.18 ON 27 NOV 1984
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM SUB_CONTR
    DEFAULTED-AS
      CHARACTERS 1
REPORT OF ITEM SUB_REM
    EDITION 1 ENCODED BY MASTER AT 19.26.19 ON 27 NOV 1984
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM SUB_REM
    DEFAULTED-AS
      CHARACTERS 20
REPORT OF ITEM SCH_AWARD_DT
    EDITION 1 ENCODED BY MASTER AT 19.26.20 ON 27 NOV 1984
    ALIAS
      SUBAWD_DTSCH
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM SCH_AWARD_DT
    DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM SUB_AWARD_DT
    EDITION 1 ENCODED BY MASTER AT 19.26.21 ON 27 NOV 1984
    ALIAS
      SUBAWD_DTACT
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM SUB_AWARD_DT
    DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM SUBCAD
    EDITION 1 ENCODED BY MASTER AT 19.26.22 ON 27 NOV 1984
    ALIAS
      SCON_AW_DT
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM SUBCAD
    DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM SUBBOD
    EDITION 1 ENCODED BY MASTER AT 19.26.23 ON 27 NOV 1984
    ALIAS
      SBEN_OCC_DT
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM SUBBOD
    DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM SUBBIDOD
    EDITION 1 ENCODED BY MASTER AT 19.26.24 ON 27 NOV 1984
    ALIAS
      SBID_OP_DT
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM SUBBIDOD
    DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM SUBNOBID
    EDITION 1 ENCODED BY MASTER AT 19.26.25 ON 27 NOV 1984
    ALIAS
      SNO_BIDDERS
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM SUBNOBID
    DEFAULTED-AS
      NUMERIC-CHARACTER 4
REPORT OF ITEM SUBGEST
    EDITION 1 ENCODED BY MASTER AT 19.26.26 ON 27 NOV 1984
    ALIAS
      SGOVT_EST
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM SUBGEST
    DEFAULTED-AS
      NUMERIC-CHARACTER 8
REPORT OF ITEM SUBLBID
    EDITION 1 ENCODED BY MASTER AT 19.26.27 ON 27 NOV 1984
```

```
   ALIAS
     SLOW_BID
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM SUBLBID
   DEFAULTED-AS
     NUMERIC-CHARACTER 8
REPORT OF ITEM SUBHBID
   EDITION 1 ENCODED BY MASTER AT 19.26.29 ON 27 NOV 1984
   ALIAS
     SUB_HIGH_BID
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM SUBHBID
   DEFAULTED-AS
     NUMERIC-CHARACTER 8
REPORT OF ITEM SAWD_CWE
   EDITION 1 ENCODED BY MASTER AT 19.26.30 ON 27 NOV 1984
   ALIAS
     SUBAWD_CWE
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM SAWD_CWE
   DEFAULTED-AS
     NUMERIC-CHARACTER 8
REPORT OF FILE MACOMSEG
   EDITION 1 ENCODED BY MASTER AT 19.27.01 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 30 DIRECT REFERENCES
   FILE MACOMSEG
   PARENT MAIN
   SEGTYPE U
   FORM DEFAULTED-AS
   CONTAINS
      ITEM  MTPN
      ITEM  MPDES
      ITEM  MPY
      ITEM  MOUS
      ITEM  MCATCD6
      ITEM  MPRI
      ITEM  MDDSORT
      ITEM  MPE
      ITEM  MSCOPE
      ITEM  MTF
      ITEM  MMISSION
      ITEM  MPA
      ITEM  MCONS
      ITEM  MRC
      ITEM  MINCODE
      ITEM  MPDIP
      ITEM  MPDIPNAME
      ITEM  MPRCD
      ITEM  MRMKI
      ITEM  MSDTG
      ITEM  MPRISDTG
      ITEM  MPASDTG
      ITEM  MPNO
      ITEM  MMOBGP
      ITEM  MMPRI
      ITEM  MOBPRISDTG
      ITEM  MLSD
      ITEM  MROD
      ITEM  MTIER
      ITEM  OMPRI
REPORT OF ITEM MTPN
   EDITION 1 ENCODED BY MASTER AT 19 26.33 ON 27 NOV 1984
   ALIAS
     M_TEMP_PN
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MTPN
   DEFAULTED-AS
     CHARACTERS 7
REPORT OF ITEM MPDES
   EDITION 1 ENCODED BY MASTER AT 19.26 34 ON 27 NOV 1984
   ALIAS
     M_PROJ_DESC
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MPDES
   DEFAULTED-AS
```

```
      CHARACTERS 26
REPORT OF ITEM MFY
   EDITION 1 ENCODED BY MASTER AT 19.26.35 ON 27 NOV 1984
   ALIAS
      M_FISCAL_YEA
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MFY
   DEFAULTED-AS
      CHARACTERS 2
REPORT OF ITEM MOUS
   EDITION 1 ENCODED BY MASTER AT 19.26.36 ON 27 NOV 1984
   ALIAS
      M_ORIG_US_SV
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MOUS
   DEFAULTED-AS
      CHARACTERS 2
REPORT OF ITEM MCATCD5
   EDITION 1 ENCODED BY MASTER AT 19.26.37 ON 27 NOV 1984
   ALIAS
      M_CATCODE
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MCATCD5
   DEFAULTED-AS
      CHARACTERS 5
REPORT OF ITEM MPRI
   EDITION 1 ENCODED BY MASTER AT 19.26.38 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MPRI
   DEFAULTED-AS
      CHARACTERS 5
REPORT OF ITEM MDDSORT
   EDITION 1 ENCODED BY MASTER AT 19.26.39 ON 27 NOV 1984
   ALIAS
      M_SORT_CODE
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MDDSORT
   DEFAULTED-AS
      CHARACTERS 2
REPORT OF ITEM MPE
   EDITION 1 ENCODED BY MASTER AT 19.26.40 ON 27 NOV 1984
   ALIAS
      M_PROG_ELE
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MPE
   DEFAULTED-AS
      CHARACTERS 6
REPORT OF ITEM MSCOPE
   EDITION 1 ENCODED BY MASTER AT 19.26.40 ON 27 NOV 1984
   ALIAS
      M_SCOPE
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MSCOPE
   DEFAULTED-AS
      NUMERIC-CHARACTER 9
REPORT OF ITEM MTF
   EDITION 1 ENCODED BY MASTER AT 19.26.41 ON 27 NOV 1984
   ALIAS
      M_TYPE_FUNDS
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MTF
   DEFAULTED-AS
      CHARACTERS 1
REPORT OF ITEM MMISSION
   EDITION 1 ENCODED BY MASTER AT 19.26.42 ON 27 NOV 1984
   ALIAS
      M_MISSION
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MMISSION
   DEFAULTED-AS
      CHARACTERS 1
REPORT OF ITEM MPA
```

```
    EDITION 1 ENCODED BY MASTER AT 19.26.43 ON 27 NOV 1984
    ALIAS
      M_PROG_AMT
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM MPA
    DEFAULTED-AS
      NUMERIC-CHARACTER 8
REPORT OF ITEM MCONS
    EDITION 1 ENCODED BY MASTER AT 19.26.44 ON 27 NOV 1984
    ALIAS
      M_CONS_TYPE
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM MCONS
    DEFAULTED-AS
      CHARACTERS 1
REPORT OF ITEM MRC
    EDITION 1 ENCODED BY MASTER AT 19.26.46 ON 27 NOV 1984
    ALIAS
      M_REPLACE_CD
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM MRC
    DEFAULTED-AS
      CHARACTERS 1
REPORT OF ITEM MINCODE
    EDITION 1 ENCODED BY MASTER AT 19.26.47 ON 27 NOV 1984
    ALIAS
      M_INCODE
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM MINCODE
    DEFAULTED-AS
      CHARACTERS 5
REPORT OF ITEM MPDIP
    EDITION 1 ENCODED BY MASTER AT 19.26.48 ON 27 NOV 1984
    ALIAS
      M_PDIP
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM MPDIP
    DEFAULTED-AS
      CHARACTERS 4
REPORT OF ITEM MPDIPNAME
    EDITION 1 ENCODED BY MASTER AT 19.26.48 ON 27 NOV 1984
    ALIAS
      MNAME
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM MPDIPNAME
    DEFAULTED-AS
      CHARACTERS 6
REPORT OF ITEM MPRCD
    EDITION 1 ENCODED BY MASTER AT 19.26.49 ON 27 NOV 1984
    ALIAS
      M_PR_CODE
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM MPRCD
    DEFAULTED-AS
      CHARACTERS 2
REPORT OF ITEM MRMKI
    EDITION 1 ENCODED BY MASTER AT 19.26.50 ON 27 NOV 1984
    ALIAS
      M_REMARK
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM MRMKI
    DEFAULTED-AS
      CHARACTERS 12
REPORT OF ITEM MSDTG
    EDITION 1 ENCODED BY MASTER AT 19.26.51 ON 27 NOV 1984
    ALIAS
      MSYS_DT_TIME
    THIS MEMBER IS DIRECTLY REFERRED TO ONCE
    THIS MEMBER CONTAINS 0 DIRECT REFERENCES
    ITEM MSDTG
    DEFAULTED-AS
      CHARACTERS 12
REPORT OF ITEM MPRISDTG
```

```
   EDITION 1 ENCODED BY MASTER AT 19.26.52 ON 27 NOV 1984
   ALIAS
     MPRISYSDT
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MPRISDTG
   DEFAULTED-AS
     CHARACTERS 12
REPORT OF ITEM MPASDTG
   EDITION 1 ENCODED BY MASTER AT 19.26.52 ON 27 NOV 1984
   ALIAS
     MPASYSDT
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MPASDTG
   DEFAULTED-AS
     CHARACTERS 12
REPORT OF ITEM MFNO
   EDITION 1 ENCODED BY MASTER AT 19.26.53 ON 27 NOV 1984
   ALIAS
     MFORMNO
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MFNO
   DEFAULTED-AS
     CHARACTERS 7
REPORT OF ITEM MMOBGP
   EDITION 1 ENCODED BY MASTER AT 19.26.54 ON 27 NOV 1984
   ALIAS
     MMOB_GROUP
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MMOBGP
   DEFAULTED-AS
     CHARACTERS 1
REPORT OF ITEM MMPRI
   EDITION 1 ENCODED BY MASTER AT 19.26.55 ON 27 NOV 1984
   ALIAS
     MMOBPRI
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MMPRI
   DEFAULTED-AS
     CHARACTERS 6
REPORT OF ITEM MOBPRISDTG
   EDITION 1 ENCODED BY MASTER AT 19.26.56 ON 27 NOV 1984
   ALIAS
     MOBSDT
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MOBPRISDTG
   DEFAULTED-AS
     CHARACTERS 12
REPORT OF ITEM MLSD
   EDITION 1 ENCODED BY MASTER AT 19.26.57 ON 27 NOV 1984
   ALIAS
     MLATE_ST_DT
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MLSD
   DEFAULTED-AS
     CHARACTERS 6
REPORT OF ITEM MROD
   EDITION 1 ENCODED BY MASTER AT 19.26.58 ON 27 NOV 1984
   ALIAS
     MRQD_OCP_DT
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MROD
   DEFAULTED-AS
     CHARACTERS 6
REPORT OF ITEM MTIER
   EDITION 1 ENCODED BY MASTER AT 19.26.59 ON 27 NOV 1984
   ALIAS
     MACOMTIER
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM MTIER
   DEFAULTED-AS
     CHARACTERS 1
REPORT OF ITEM OMPRI
```

```
   EDITION 1 ENCODED BY MASTER AT 19.27.00 ON 27 NOV 1984
   ALIAS
     OTHERMPRI
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM OMPRI
   DEFAULTED-AS
     CHARACTERS 6
REPORT OF FILE HISTZCP1
   EDITION 1 ENCODED BY MASTER AT 19.27.07 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 3 DIRECT REFERENCES
   FILE HISTZCP1
   PARENT MAIN
   SEGTYPE SH1
   SORT-KEY
     ITEM  HIS_ZCP1_DT DESCENDING
   FORM DEFAULTED-AS
   CONTAINS
     ITEM  HIS_ZCP1_DT
     ITEM  HIS_PROG_MAT
REPORT OF ITEM HIS_ZCP1_DT
   EDITION 1 ENCODED BY MASTER AT 19.27.05 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO 2 TIMES
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_ZCP1_DT
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM HIS_PROG_MAT
   EDITION 1 ENCODED BY MASTER AT 19.27.06 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_PROG_MAT
   DEFAULTED-AS
     NUMERIC-CHARACTER 8
REPORT OF FILE HISTZCP2
   EDITION 1 ENCODED BY MASTER AT 19.27.11 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 5 DIRECT REFERENCES
   FILE HISTZCP2
   PARENT MAIN
   SEGTYPE SH1
   SORT-KEY
     ITEM  HIS_ZCP2_DT DESCENDING
   FORM DEFAULTED-AS
   CONTAINS
     ITEM  HIS_ZCP2_DT
     ITEM  HIS_ZCP_DR
     ITEM  HIS_ZCP_DCD
     ITEM  HZP2_FILLER
REPORT OF ITEM HIS_ZCP2_DT
   EDITION 1 ENCODED BY MASTER AT 19.27.08 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO 2 TIMES
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_ZCP2_DT
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM HIS_ZCP_DR
   EDITION 1 ENCODED BY MASTER AT 19.27.09 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_ZCP_DR
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM HIS_ZCP_DCD
   EDITION 1 ENCODED BY MASTER AT 19.27.10 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_ZCP_DCD
   DEFAULTED-AS
     CHARACTERS 1
REPORT OF ITEM HZP2_FILLER
   EDITION 1 ENCODED BY MASTER AT 19.27.11 ON 27 NOV 1984
   ALIAS
     FILL2
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HZP2_FILLER
   DEFAULTED-AS
     CHARACTERS 3
REPORT OF FILE HISTZCP3
```

```
   EDITION 1 ENCODED BY MASTER AT 19.27.18 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 6 DIRECT REFERENCES
   FILE HISTZCP3
   PARENT MAIN
   SEGTYPE SH1
   SORT-KEY
      ITEM  HIS_ZCP3_DT DESCENDING
   FORM DEFAULTED-AS
   CONTAINS
      ITEM  HIS_ZCP3_DT
      ITEM  HIS_2807_CNG
      ITEM  HIS_2807_OSD
      ITEM  HIS_2807_RQ
      ITEM  HZP3_FILLER
REPORT OF ITEM HIS_ZCP3_DT
   EDITION 1 ENCODED BY MASTER AT 19.27.14 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO 2 TIMES
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_ZCP3_DT
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM HIS_2807_CNG
   EDITION 1 ENCODED BY MASTER AT 19.27.14 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_2807_CNG
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM HIS_2807_OSD
   EDITION 1 ENCODED BY MASTER AT 19.27.15 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_2807_OSD
   DEF807_RQ
   EDITION 1 ENCODED BY MASTER AT 19.27.16 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_2807_RQ
   DEFAULTED-AS
      CHARACTERS 1
REPORT OF ITEM HZP3_FILLER
   EDITION 1 ENCODED BY MASTER AT 19.27.17 ON 27 NOV 1984
   ALIAS
      FILL13
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HZP3_FILLER
   DEFAULTED-AS
      CHARACTERS 3
REPORT OF FILE HISTZCP4
   EDITION 1 ENCODED BY MASTER AT 19.27.24 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 6 DIRECT REFERENCES
   FILE HISTZCP4
   PARENT MAIN
   SEGTYPE SH1
   SORT-KEY
      ITEM  HIS_ZCP4_DT DESCENDING
   FORM DEFAULTED-AS
   CONTAINS
      ITEM  HIS_ZCP4_DT
      ITEM  HIS_1391_REC
      ITEM  HIS_1391_DIS
      ITEM  HIS_1391_OK
      ITEM  HZP4_FILLER
REPORT OF ITEM HIS_ZCP4_DT
   EDITION 1 ENCODED BY MASTER AT 19.27.20 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO 2 TIMES
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_ZCP4_DT
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM HIS_1391_REC
   EDITION 1 ENCODED BY MASTER AT 19.27.20 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_1391_REC
   DEFAULTED-AS
      NUMERIC-CHARACTER 6
REPORT OF ITEM HIS_1391_DIS
```

```
   EDITION 1 ENCODED BY MASTER AT 19.27.21 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_1391_DIS
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM HIS_1391_OK
   EDITION 1 ENCODED BY MASTER AT 19.27.22 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_1391_OK
   DEFAULTED-AS
     CHARACTERS 1
REPORT OF ITEM HZP4_FILLER
   EDITION 1 ENCODED BY MASTER AT 19.27.23 ON 27 NOV 1984
   ALIAS
     FILL14
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HZP4_FILLER
   DEFAULTED-AS
     CHARACTERS 3
REPORT OF FILE HISTZCP6
   EDITION 1 ENCODED BY MASTER AT 19.27.27 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 3 DIRECT REFERENCES
   FILE HISTZCP6
   PARENT MAIN
   SEGTYPE SH1
   SORT-KEY
     ITEM  HIS_ZCP6_DT DESCENDING
   FORM DEFAULTED-AS
   CONTAINS
     ITEM  HIS_ZCP6_DT
     ITEM  HIS_DSE_SD
REPORT OF ITEM HIS_ZCP6_DT
   EDITION 1 ENCODED BY MASTER AT 19.27.26 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO 2 TIMES
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_ZCP6_DT
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM HIS_DSE_SD
   EDITION 1 ENCODED BY MASTER AT 19.27.27 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_DSE_SD
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF FILE HISTACE1
   EDITION 1 ENCODED BY MASTER AT 19.27.45 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 16 DIRECT REFERENCES
   FILE HISTACE1
   PARENT MAIN
   SEGTYPE SH1
   SORT-KEY
     ITEM  HATRANS DESCENDING
   FORM DEFAULTED-AS
   CONTAINS
     ITEM  HATRANS
     ITEM  HKN
     ITEM  HACTDR
     ITEM  HACEDR
     ITEM  HACEDCD
     ITEM  HPA
     ITEM  HFY
     ITEM  HSCP
     ITEM  HPFY
     ITEM  HDTFY
     ITEM  HDTPA
     ITEM  HODRCN
     ITEM  HNDRCN
     ITEM  HPDRCN
     ITEM  HACEFILLER
REPORT OF ITEM HATRANS
   EDITION 1 ENCODED BY MASTER AT 19.27.29 ON 27 NOV 1984
   ALIAS
     ACE1TRANS
   THIS MEMBER IS DIRECTLY REFERRED TO 2 TIMES
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
```

```
   ITEM HATRANS
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM HKN
   EDITION 1 ENCODED BY MASTER AT 19.27.30 ON 27 NOV 1984
   ALIAS
     HISKEYNR
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HKN
   DEFAULTED-AS
     CHARACTERS 13
REPORT OF ITEM HACTDR
   EDITION 1 ENCODED BY MASTER AT 19.27.31 ON 27 NOV 1984
   ALIAS
     ACTDR
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HACTDR
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM HACEDR
   EDITION 1 ENCODED BY MASTER AT 19.27.32 ON 27 NOV 1984
   ALIAS
     HISACEDR
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HACEDR
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM HACEDCD
   EDITION 1 ENCODED BY MASTER AT 19.27.32 ON 27 NOV 1984
   ALIAS
     HISACEDCD
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HACEDCD
   DEFAULTED-AS
     CHARACTERS 1
REPORT OF ITEM HPA
   EDITION 1 ENCODED BY MASTER AT 19.27.33 ON 27 NOV 1984
   ALIAS
     HISPADES
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HPA
   DEFAULTED-AS
     NUMERIC-CHARACTER 8
REPORT OF ITEM HFY
   EDITION 1 ENCODED BY MASTER AT 19.27.34 ON 27 NOV 1984
   ALIAS
     HISFYDES
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HFY
   DEFAULTED-AS
     CHARACTERS 2
REPORT OF ITEM HSCP
   EDITION 1 ENCODED BY MASTER AT 19.27.35 ON 27 NOV 1984
   ALIAS
     HISSCOPEDES
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HSCP
   DEFAULTED-AS
     NUMERIC-CHARACTER 9
REPORT OF ITEM HPFY
   EDITION 1 ENCODED BY MASTER AT 19.27.36 ON 27 NOV 1984
   ALIAS
     HISPCFY
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HPFY
   DEFAULTED-AS
     CHARACTERS 2
REPORT OF ITEM HDTFY
   EDITION 1 ENCODED BY MASTER AT 19.27.38 ON 27 NOV 1984
   ALIAS
     HISDATEFY
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
```

```
   ITEM HDTFY
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM HDTPA
   EDITION 1 ENCODED BY MASTER AT 19.27.39 ON 27 NOV 1984
   ALIAS
     HISDATEPA
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HDTPA
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM HODRCN
   EDITION 1 ENCODED BY MASTER AT 19.27.41 ON 27 NOV 1984
   ALIAS
     HISOLDDRCN
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HODRCN
   DEFAULTED-AS
     CHARACTERS 3
REPORT OF ITEM HNDRCN
   EDITION 1 ENCODED BY MASTER AT 19.27.42 ON 27 NOV 1984
   ALIAS
     HISNEWDRCN
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HNDRCN
   DEFAULTED-AS
     CHARACTERS 3
REPORT OF ITEM HPDRCN
   EDITION 1 ENCODED BY MASTER AT 19.27.43 ON 27 NOV 1984
   ALIAS
     HISPREDRCN
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HPDRCN
   DEFAULTED-AS
     CHARACTERS 3
REPORT OF ITEM HACEFILLER
   EDITION 1 ENCODED BY MASTER AT 19.27.44 ON 27 NOV 1984
   ALIAS
     FILL15
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HACEFILLER
   DEFAULTED-AS
     CHARACTERS 1
REPORT OF FILE HISTMPES
   EDITION 1 ENCODED BY MASTER AT 19.27.53 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 7 DIRECT REFERENCES
   FILE HISTMPES
   PARENT MAIN
   SEGTYPE SH1
   SORT-KEY
     ITEM  HIS_MPES_DT DESCENDING
   FORM DEFAULTED-AS
   CONTAINS
     ITEM  HIS_MPES_DT
     ITEM  HIS_CST_DT
     ITEM  HIS_PRJ_CST
     ITEM  HIS_CST_CD
     ITEM  HIS_SOLAR
     ITEM  HMPS_FILLER
REPORT OF ITEM HIS_MPES_DT
   EDITION 1 ENCODED BY MASTER AT 19.27.48 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO 2 TIMES
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_MPES_DT
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM HIS_CST_DT
   EDITION 1 ENCODED BY MASTER AT 19.27.49 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_CST_DT
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM HIS_PRJ_CST
   EDITION 1 ENCODED BY MASTER AT 19.27.50 ON 27 NOV 1984
```

```
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_PRJ_CST
   DEFAULTED-AS
     NUMERIC-CHARACTER 8
REPORT OF ITEM HIS_CST_CD
   EDITION 1 ENCODED BY MASTER AT 19.27.50 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_CST_CD
   DEFAULTED-AS
     CHARACTERS 1
REPORT OF ITEM HIS_SOLAR
   EDITION 1 ENCODED BY MASTER AT 19.27.51 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_SOLAR
   DEFAULTED-AS
     NUMERIC-CHARACTER 8
REPORT OF ITEM HMPS_FILLER
   EDITION 1 ENCODED BY MASTER AT 19.27.52 ON 27 NOV 1984
   ALIAS
     FILL16
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HMPS_FILLER
   DEFAULTED-AS
     CHARACTERS 3
REPORT OF FILE HISTMPE1
   EDITION 1 ENCODED BY MASTER AT 19.28.01 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 8 DIRECT REFERENCES
   FILE HISTMPE1
   PARENT MAIN
   SEGTYPE SH1
   SORT-KEY
     ITEM  HIS_MPE1_DT DESCENDING
   FORM DEFAULTED-AS
   CONTAINS
     ITEM  HIS_MPE1_DT
     ITEM  HIS_MPE_DR
     ITEM  HIS_DEI_DT
     ITEM  HIS_DEI_RQ
     ITEM  HIS_MPE_DCD
     ITEM  HISENREVDT
     ITEM  HISENRN
REPORT OF ITEM HIS_MPE1_DR
   EDITION 1 ENCODED BY MASTER AT 19.27.55 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO 2 TIMES
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_MPE1_DT
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM HIS_MPE_DR
   EDITION 1 ENCODED BY MASTER AT 19.27.56 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_MPE_DR
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM HIS_DEI_DT
   EDITION 1 ENCODED BY MASTER AT 19.27.57 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_DEI_DT
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM HIS_MPE_DQ
   EDITION 1 ENCODED BY MASTER AT 19.27.58 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_DEI_RQ
   DEFAULTED-AS
     CHARACTERS 1
REPORT OF ITEM HIS_MPE_DCD
   EDITION 1 ENCODED BY MASTER AT 19.27.59 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_MPE_DCD
   DEFAULTED-AS
     CHARACTERS 1
```

```
REPORT OF ITEM HISENREVDT
  EDITION 1 ENCODED BY MASTER AT 19.27.59 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM HISENREVDT
  DEFAULTED-AS
    NUMERIC-CHARACTER 6
REPORT OF ITEM HISENRN
  EDITION 1 ENCODED BY MASTER AT 19.28.00 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM HISENRN
  DEFAULTED-AS
    CHARACTERS 2
REPORT OF FILE HISTMPC1
  EDITION 1 ENCODED BY MASTER AT 19.28.07 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 5 DIRECT REFERENCES
  FILE HISTMPC1
  PARENT MAIN
  SEGTYPE SH1
  SORT-KEY
    ITEM  HIS_MPC1_DT DESCENDING
  FORM DEFAULTED-AS
  CONTAINS
    ITEM  HIS_MPC1_DT
    ITEM  HIS_DES_DD
    ITEM  HIS_MPC_DCD
    ITEM  HMPC_FILLER
REPORT OF ITEM HIS_MPC1_DT
  EDITION 1 ENCODED BY MASTER AT 19.28.04 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO 2 TIMES
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM HIS_MPC1_DT
  DEFAULTED-AS
    NUMERIC-CHARACTER 6
REPORT OF ITEM HIS_DES_DD
  EDITION 1 ENCODED BY MASTER AT 19.28.05 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM HIS_DES_DD
  DEFAULTED-AS
    NUMERIC-CHARACTER 6
REPORT OF ITEM HIS_MPC_DCD
  EDITION 1 ENCODED BY MASTER AT 19.28.06 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM HIS_MPC_DCD
  DEFAULTED-AS
    CHARACTERS 1
REPORT OF ITEM HMPC_FILLER
  EDITION 1 ENCODED BY MASTER AT 19.28.06 ON 27 NOV 1984
  ALIAS
    FILL18
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM HMPC_FILLER
  DEFAULTED-AS
    CHARACTERS 3
REPORT OF FILE HISTMPC2
  EDITION 1 ENCODED BY MASTER AT 19.28.12 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 4 DIRECT REFERENCES
  FILE HISTMPC2
  PARENT MAIN
  SEGTYPE SH1
  SORT-KEY
    ITEM  HIS_MPC2_DT DESCENDING
  FORM DEFAULTED-AS
  CONTAINS
    ITEM  HIS_MPC2_DT
    ITEM  HIS_CONCOM
    ITEM  HIS_DES_AMT
REPORT OF ITEM HIS_MPC2_DT
  EDITION 1 ENCODED BY MASTER AT 19.28.09 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO 2 TIMES
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
  ITEM HIS_MPC2_DT
  DEFAULTED-AS
    NUMERIC-CHARACTER 6
REPORT OF ITEM HIS_CONCOM
```

```
   EDITION 1 ENCODED BY MASTER AT 19.28.10 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_CONCOM
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM HIS_DES_AMT
   EDITION 1 ENCODED BY MASTER AT 19.28.11 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_DES_AMT
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF FILE HISTLVL
   EDITION 1 ENCODED BY MASTER AT 19.28.17 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 4 DIRECT REFERENCES
   FILE HISTLVL
   PARENT MAIN
   SEGTYPE SHI
   SORT-KEY
     ITEM  HIS_DT_LVL DESCENDING
   FORM DEFAULTED-AS
   CONTAINS
     ITEM  HIS_DT_LVL
     ITEM  HIS_LVL
     ITEM  HLVL_FILLER
REPORT OF ITEM HIS_DT_LVL
   EDITION 1 ENCODED BY MASTER AT 19.28.14 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO 2 TIMES
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_DT_LVL
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM HIS_LVL
   EDITION 1 ENCODED BY MASTER AT 19.28.15 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_LVL
   DEFAULTED-AS
     CHARACTERS 1
REPORT OF ITEM HLVL_FILLER
   EDITION 1 ENCODED BY MASTER AT 19.28.16 ON 27 NOV 1984
   ALIAS
     FILL19
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HLVL_FILLER
   DEFAULTED-AS
     CHARACTERS 3
REPORT OF FILE HISTCFY
   EDITION 1 ENCODED BY MASTER AT 19.28.20 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 3 DIRECT REFERENCES
   FILE HISTCFY
   PARENT MAIN
   SEGTYPE SHI
   SORT-KEY
     ITEM  HIS_DT_CFY DESCENDING
   FORM DEFAULTED-AS
   CONTAINS
     ITEM  HIS_DT_CFY
     ITEM  HIS_CFY
REPORT OF ITEM HIS_DT_CFY
   EDITION 1 ENCODED BY MASTER AT 19.28.19 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO 2 TIMES
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_DT_CFY
   DEFAULTED-AS
     NUMERIC-CHARACTER 6
REPORT OF ITEM HIS_CFY
   EDITION 1 ENCODED BY MASTER AT 19.28.20 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
   ITEM HIS_CFY
   DEFAULTED-AS
     CHARACTERS 4
REPORT OF FILE COM2SEG
   EDITION **DUMMY** CREATED BY MASTER AT 19.28.36 ON 27 NOV 1984
   THIS MEMBER IS DIRECTLY REFERRED TO ONCE
   THIS MEMBER CONTAINS 0 DIRECT REFERENCES
```

```
REPORT OF FILE SITESEG
  EDITION **DUMMY** CREATED BY MASTER AT 19.28.36 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
REPORT OF FILE CATSEG
  EDITION **DUMMY** CREATED BY MASTER AT 19.28.36 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
REPORT OF FILE CAT3DESC
  EDITION **DUMMY** CREATED BY MASTER AT 19.28.36 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
REPORT OF FILE DESCRSEG
  EDITION **DUMMY** CREATED BY MASTER AT 19.28.36 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
REPORT OF FILE DIVSEG
  EDITION **DUMMY** CREATED BY MASTER AT 19.28.36 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
REPORT OF FILE INSTSEG
  EDITION **DUMMY** CREATED BY MASTER AT 19.28.36 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
REPORT OF FILE MACMSEG
  EDITION **DUMMY** CREATED BY MASTE   T 19.28.36 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO  NCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
REPORT OF FILE ZBDICSEG
  EDITION **DUMMY** CREATED BY MASTER AT 19.28.36 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
REPORT OF FILE AUTHSEG
  EDITION **DUMMY** CREATED BY MASTER AT 19.28.36 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
REPORT OF FILE NOTEAUTH
  EDITION **DUMMY** CREATED BY MASTER AT 19.28.36 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
REPORT OF FILE APPRSEG
  EDITION **DUMMY** CREATED BY MASTER AT 19.28.36 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
REPORT OF FILE NOTESEG
  EDITION **DUMMY** CREATED BY MASTER AT 19.28.36 ON 27 NOV 1984
  THIS MEMBER IS DIRECTLY REFERRED TO ONCE
  THIS MEMBER CONTAINS 0 DIRECT REFERENCES
>enddmr
DM000621                END OF DATA ON PRIMARY INPUT FILE
DM000721                      642 FREE BLOCKS 038 FREE BLOCKS ON DATA ENTRIES DATA SET
DM000731                      966 FREE BLOCKS ON INDEX DATA SET
DM000761                        2 % UTILIZATION OF ERROR RECOVERY DATA SET
DM000801                     8480 BYTES USED IN STACK STORAGE
DM000811                    63688 BYTES USED IN NON-STACK STORAGE
DM000701                        4 LINES READ FROM PRIMARY INPUT FILE
DM000711                     3360 LINES PRINTED
R; T=5 91/16 61 15 28 16
```

APPENDIX C:

DATA USED TO IDENTIFY STANDARD REPORTS

The Standard Report Definition file contains the information shown in Tables C1 through C3.

REPORT ID	TARGET ITEM NAME	QUALIFYING ITEM NAME
1	prog_ele	fy
2	auth_amt,prog_amt,cwe_amt, des_percent, des_st_dt, des_comp_dt,dist_name	fy

Table C1. Target data items and qualifying items for two reports.

DATA ITEM NAME	INTERNAL KEYWORD
prog_ele	amount,element,program
auth_amt	amount,authorized,authority
prog_amt	amount,program
cwe_amt	current,estimate,amount,project
des_percent	progress,design,completion,percent
des_st_dt	design,start,date
des_comp_dt	design,completion,date
dist_name	district,title,regional
fy	current,year,congress,fiscal

Table C2. Internal keywords corresponding to data items in Table C1.

INTERNAL KEYWORD	SEARCH KEYWORD
program,project	program,project
current	current
amount	amount,cost
fiscal	fiscal
year	year,yr
element	element
congress	congress
estimate	estimate
design	design
completion	completion
percent	percent
start	start
date	date
district	district
title	title
regional	regional,region
authority	authoirty
authorized	authorized
progress	progress

Table C3. Contents of the current TSK.

APPENDIX D:

SAMPLE QUERY PROCESSING SESSION

```
CMS>stdrep

                **** WELCOME TO THE REPORT GENERATOR ****

        THIS SYSTEM IS A QUESTION - DRIVEN INTERACTIVE ONE.
        IT ACCEPTS AN INITIAL QUERY FROM THE USER, E.G.,
           SHOW THE COST FOR PROJECT=B12 AND YEAR = 81
        AND DECIDES WHICH REPORTS SATISFY THE INITIAL QUERY
        AND THEN GENERATES THE REPORT SELECTED BY THE USER.

        JUST FOLLOW THE PROMPTS AND PROCEED THROUGH TO
        OBTAIN THE DESIRED REPORT.

        NOTE:   THE REPORT GENERATOR MAY BE EXITED AT ANY
                TIME A PROMPT IS PRESENTED BY TYPING "QUIT"
                 EXCEPT WHEN PROMPTED WITH
                "ENTER INITIAL QUERY", THEN EITHER "ENDQ"
                 OR A NULL LINE ARE REQUIRED TO EXIT.


ENTER AN INITIAL QUERY

>tell me about cost summarized by program element

:
>

FOLLOWING STANDARD REPORT(S) SATISFY YOUR QUERY
  REPORT ID   REPORT FULL NAME
.................................................
           1   SUMELEM

WHICH OF THE ABOVE REPORT DO YOU WANT?
        TYPE WANTED REPORT ID (E.G.1,2,... )
        OR TYPE ? FOR MORE DESCRIPTION
        OF THE REPORT(S)
        OR, TYPE RETURN KEY IF YOU DON'T LIKE ANY
        OF THESE REPORTS
>?

  REPORT ID          REPORT FULL NAME          REPORT DESCRIPTION
.....................................................................
        1      SUMELEM                         SUMMARY BY PROGRAM ELEMEN
                                               T DOLLARS IN THOUSANDS


WHICH OF THE ABOVE REPORT DO YOU WANT?
        TYPE WANTED REPORT ID (E.G.1,2,....)
        OR, TYPE RETURN KEY IF YOU DON'T LIKE ANY
        OF THESE REPORTS
>1

YOU HAVE SELECTED THE FOLLOWING STANDARD REPORTS
FOR GENERATION

  REPORT ID   REPORT FULL NAME
.................................................
           1   SUMELEM
IS THIS LISTING CORRECT ?
Type YES/NO
>yes



TO COMPLETE THE QUERY, YOU MUST SPECIFY THE VALUES OF FOLLOWING DATA ITEMS
CFY            IN PMMFILE  FORMAT - A2
DESC FISCAL YEAR

>84

WORKING X   47 ON 12/02/84
```

```
FOCUS   4.0.14V <ST>     CREATED 01/26/84
  >>>>>>>>>>
WELCOME TO FOCUS

>>>>>>>>>>>>WORKING
 NUMBER OF RECORDS IN TABLE=     376   LINES=      6

 PAUSE.. PLEASE ISSUE CARRIAGE RETURN WHEN READY
>

 PAGE      1

          FY 1984  SUMMARY BY PROGRAM ELEMENT
                DOLLARS IN THOUSANDS
                                          FY
                                          84
 PRO_TITLE
 ---------------------------------------------------
 GENERAL PURPOSE FORCES                      768,982
 INTELLIGENCE                                  8,936
 RESEARCH & DEVELOPMENT                       59,284
 CENTRAL SUPPLY & MAINT                       73,871
 TRAINING & PERSONNEL                        182,002
 ADMINISTRATIVE                              209,666

 TOTAL                                     1,302,741


>SELECT ACTION TO BE TAKEN:

        1  EXECUTE A DIFFERENT QUERY.
        2  EXIT FROM THE REPORT GENERATOR.

>1

              **** WELCOME TO THE REPORT GENERATOR ****


       THIS SYSTEM IS A QUESTION - DRIVEN INTERACTIVE ONE.
       IT ACCEPTS AN INITIAL QUERY FROM THE USER, E.G.,
          SHOW THE COST FOR PROJECT=B12 AND YEAR = 81
       AND DECIDES WHICH REPORTS SATISFY THE INITIAL QUERY
       AND THEN GENERATES THE REPORT SELECTED BY THE USER.

       JUST FOLLOW THE PROMPTS AND PROCEED THROUGH TO
       OBTAIN THE DESIRED REPORT.

       NOTE:  THE REPORT GENERATOR MAY BE EXITED AT ANY
              TIME A PROMPT IS PRESENTED BY TYPING "QUIT"
               EXCEPT WHEN PROMPTED WITH
              "ENTER INITIAL QUERY", THEN EITHER "ENDQ"
               OR A NULL LINE ARE REQUIRED TO EXIT.


ENTER AN INITIAL QUERY
:
>tell me anything about current program cost

:
>

FOLLOWING STANDARD REPORT(S) SATISFY YOUR QUERY
  REPORT ID   REPORT FULL NAME
-----------------------------------------
          2   CONFUN2

WHICH OF THE ABOVE REPORT DO YOU WANT?
        TYPE WANTED REPORT ID (E.G. 1,2,...)
        OR TYPE ? FOR MORE DESCRIPTION
        OF THE REPORT(S)
        OR, TYPE RETURN KEY IF YOU DON'T LIKE ANY
        OF THESE REPORTS
>?

  REPORT ID          REPORT FULL NAME          REPORT DESCRIPTION
-------------------------------------------------------------------------
        2      CONFUN2                         MCA CONSTRUCTION FUNDS ST
                                               ATUS
```

```
WHICH OF THE ABOVE REPORT DO YOU WANT?
        TYPE WANTED REPORT ID (E.G. 1,2,....)
        OR, TYPE RETURN KEY IF YOU DON'T LIKE ANY
        OF THESE REPORTS
>


SELECT ACTION TO BE TAKEN:

        1. EXECUTE A DIFFERENT QUERY.
        2. EXIT FROM THE REPORT GENERATOR.

>2

16:49 EDT
CMS>
```

APPENDIX E:

LISTING OF FILES

```
YSIMAP:
  PROCEDURE OPTIONS(MAIN);


/*
 *  PROJECT:          REPORT GENERATOR
 *  PROGRAMMER:       DEPARTMENT OF COMPUTER SCIENCE
 *                    UNIVERSITY OF ILLINOIS
 *
 *  FILENAME:         THESAURUS.PLI
 *  LANGUAGE:         PL/I UNDER CMS
 *  DEPENDENCIES:     SKWDIN DATA
 *
 *  DATE              DECEMBER, 1984
 */


/*  THIS MODULE READS IN ALL THE USER SEARCH KEYWORDS AND THEIR
    ASSOCIATED INTERNAL KEYWORDS FROM THE DATAFILE SKWDIN.  IT
    THEN CREATES THE HASH TABLE AND WRITED THE TABLE IN TO
    THE DATA FILE SKWDOUT FOR USE DURING QUERY PARSING
*/


  DCL SKWDOUT FILE OUTPUT STREAM ENV (F RECSIZE(80));
  DCL NULL BUILTIN;
  DCL WDLENG DEC FIXED(2) INIT(12);       /* WORD LENGTH */
  DCL (SKWD,INKWD) CHAR(12) VARYING,      /* USER SEARCH KEYWORD,  */
                                          /* INTERNAL KEYWORD */
       WD CHAR(12) VARYING ,              /* WORD OR TOKEN */
       END_LINE CHAR(12) VARYING INIT(';'),
       MORE_REC BIT(1) INIT ('1'B),       /* MORE RECORDS IN FILE */
       YES BIT(1) INIT('1'B),
       NO BIT(1) INIT('0'B),
       NEXT DEC FIXED INIT(1012),
       FLAG BIT(1),
       INFL BIT(1),
       INDX DEC FIXED,
       (I,J,K) DEC FIXED,
       MORE_WD BIT(1),                    /* MORE WORDS IN LINE */
       HTBSIZE DEC FIXED INIT(1213),      /* HASH TABLE SIZE */
       BKSIZE DEC FIXED INIT(4),          /* BUCKET SIZE */
       TBSIZE DEC FIXED INIT(1013),       /* OVERFLOW TABLE SIZE */
       C(80) CHAR(1),                     /* INPUT LINE BUFFER */
       PTR POINTER,
  DCL 1 HTABLE(0:HTBSIZE-1),              /* HASH TABLE -- THESAURUS */
         2 TBL(BKSIZE),
           3 SKWDF CHAR(12),              /* USER SEARCH KEYWORD */
           3 COUNT DEC FIXED INIT((HTBSIZE*BKSIZE)0),
                                          /* NUMBER INTERNAL KEYWORDS */
           3 INKWDPT POINTER,             /* TO INTERNAL KEYWORD LIST */
      2 OVERFL DEC FIXED INIT((HTBSIZE)0);
  DCL 1 INKWDREC BASED(CURR_INKWD_PT),    /* NODE OF INTERNAL KEYWORD LIST */
         2 INKWDF CHAR(12) VARYING,       /* INTERNAL KEYWORD */
         2 NEXTINKWD POINTER;             /* NEXT INTERNAL KEYWORD */


/*********************/
/*  MAIN PROCEDURE  */
/*********************/

  ON ENDFILE(SYSIN) MORE_REC=NO;
  DO WHILE(MORE_REC);
       GET EDIT(C) (80 A(1));
       INFL=YES;
       MORE_WD=MORE_REC;
     DO WHILE(MORE_WD);
       WD=GETWD;
       IF (WD = '') THEN MORE_WD=NO;
     ELSE DO;
       IF INFL THEN
     DO;
       INKWD=WD;
       INFL=NO;
     END;
```

```
      ELSE DO;
        SKWD=WD;
            /* COMPUTE HASHING FUNCTION INDEX */
            INDX=MOD(HASHF(SKWD),TBSIZE);
            FLAG=YES;
            DO WHILE (FLAG);
                I=1;
                /* CHECK IF BUCKET IS FULL */
                DO WHILE (FLAG & (I<=BKSIZE));
                    IF (COUNT(INDX,I) NE 0) THEN DO;
/* CHECK IF THE RECORD IS OCCUPIED */
                        IF (SKWDF(INDX,I)=SKWD) THEN
/* CHECK IF THE SEARCH KEYWORD IS THE SAME */
                            DO;
                                PTR=INKWDPT(INDX,I);
/* INSERT INTERNAL KEYWORD*/
                                DO WHILE (FLAG & (PTR NE NULL));
                                    IF PTR->INKWDF=INKWD THEN DO;
                                        PUT LIST('DUPLICATED',INKWD);
                                        FLAG=NO;
                                    END; /* END IF PTR-> THEN */
                                    ELSE DO;
                                        PTR=PTR->NEXTINKWD;
                                    END; /* END ELSE */
                                END; /* END WHILE */
                                IF FLAG THEN DO;
                                    COUNT(INDX,I)=COUNT(INDX,I)+1;
                                    ALLOCATE INKWDREC;
                                    NEXTINKWD=INKWDPT(INDX,I);
                                    INKWDPT(INDX,I)=CURR_INKWD_PT;
                                    INKWDF=INKWD;
                                    FLAG=NO;
                                END; /* END FLAG */
                            END;
                        ELSE
                            I=I+1;
                    END;
                    ELSE DO;
                        SKWDF(INDX,I)=SKWD;
                        COUNT(INDX,I)=COUNT(INDX,I)+1;
                        ALLOCATE INKWDREC;
                        INKWDPT(INDX,I)=CURR_INKWD_PT;
                        INKWDF=INKWD;
                        NEXTINKWD=NULL;
                        FLAG=NO;
                    END;
                END;
                IF FLAG THEN
                DO;
                    IF (OVERFL(INDX)=0) THEN
                    DO;
                        NEXT=NEXT+1;
                        OVERFL(INDX)=NEXT;
                     END;
                    INDX=OVERFL(INDX);
                END;
            END; /* DO WHILE FLAG */
        END; /* END ELSE */
    END;
    END; /* DO WHILE MORE_WD */
  END; /* DO WHILE MORE_REC */
  DO I=0 TO HTBSIZE-1;
     IF (COUNT(I,1) NE 0) THEN
     DO;
     DO J=1 TO BKSIZE;
        IF (COUNT(I,J) NE 0) THEN DO;
           PUT FILE(SKWDOUT) EDIT(I,J,SKWDF(I,J),COUNT(I,J))
                             (COL(1),2(F(5),X(1)),A(12),X(1),F(5));
           PUT FILE(SKWDOUT) EDIT('') (COL(1),A(0));
           PTR=INKWDPT(I,J);
           DO K=1 TO COUNT(I,J);
              PUT FILE(SKWDOUT) (PTR->INKWDF) ;
              PTR=PTR->NEXTINKWD;
           END; /* DO K */
        END; /* IF (COUNT(I,J) THEN */
        END; /* END DO J*/
       IF (OVERFL(I) NE 0) THEN
        PUT FILE(SKWDOUT) EDIT(I,'0',OVERFL(I)) (COL(1),3 (F(5),X(1)));
     END; /* END THEN */
   END; /* END DO I */
```

```
/*****************************************************************/
/*                                                               */
/*   PROCEDURE:   GETWD                                          */
/*                                                               */
/*   PURPOSE:   THIS FUNCTION DETERMINES AND RETURNS THE         */
/*              NEXT WORD OR TOKEN FROM THE INPUT BUFFER,        */
/*              C.  IF NO 'NEXT' WORD EXISTS, THE NULL           */
/*              STRING IS RETURNED.                              */
/*                                                               */
/*****************************************************************/

  GETWD:
   PROCEDURE RETURNS(CHAR(12) VARYING);
  DCL FLAG BIT(1),
     WD CHAR(12) VARYING,
       I DEC FIXED STATIC INIT(1);
  FLAG=YES;
  DO WHILE (I LE 80 AND FLAG) ;
      IF (C(I) NE ' ') THEN
         FLAG=NO;
         ELSE I=I+1;
  END;
  WD='';
  IF (NOT FLAG) THEN
   DO;
      FLAG=YES;
      DO WHILE (I LE 80 AND FLAG);
        IF (C(I)=' ') THEN FLAG=NO;
      ELSE DO;
        WD=WD CAT C(I);
        I=I+1;
      END;
   END;
  END;
  IF (WD ='') THEN I=1;
    RETURN(WD);
  END GETWD;


/*****************************************************************/
/*                                                               */
/*   PROCEDURE:   CONV                                           */
/*                                                               */
/*   PURPOSE:   THIS FUNCTION CONVERTS A CHARACTER TO            */
/*              ITS INTERNAL NUMERICAL REPRESENTATION.           */
/*                                                               */
/*****************************************************************/

  CONV:
   PROCEDURE(CHR) RETURNS(DEC FIXED(3,0));
  DCL CHR CHAR(1);
  RETURN(UNSPEC(CHR));
  END CONV;


/*****************************************************************/
/*                                                               */
/*   PROCEDURE:   HASHF                                          */
/*                                                               */
/*   PURPOSE:   THIS FUNCTION CALCULATES AND RETURNS THE         */
/*              INDEX INTO THE HASH TABLE FOR THE                */
/*              SPECIFIED WORD.  (THE DETAILS OF THE             */
/*              HASHING FUNCTION USED CAN BE FOUND IN            */
/*              THE EXTERNAL DOCUMENTATION.)                     */
/*                                                               */
/*****************************************************************/

 HASHF:
  PROCEDURE(STR) RETURNS (DEC FIXED(12));
  DCL IND DEC FIXED(3,0),
      STR CHAR(12) VARYING,
      POS DEC FIXED(2) INIT(7) STATIC ,
      LNG DEC FIXED(2,0);
  LNG=LENGTH(STR);
  IF (LNG=0) THEN
     RETURN(0);
  ELSE
     RETURN(HASHF(SUBSTR(STR,1,LNG-1))*POS+CONV(SUBSTR(STR,LNG,1)));
  END HASHF;

  END YSIMAP;
```

```
ZSRDEF: PROCEDURE OPTIONS(MAIN);

/*
 *  PROJECT:          REPORT GENERATOR
 *  PROGRAMMER:       DEPARTMENT OF COMPUTER SCIENCE
 *                    UNIVERSITY OF ILLINOIS
 *
 *  FILENAME:         DFNGENERATOR.PLI
 *  LANGUAGE:         PL/I UNDER CMS
 *  DEPENDENCIES:
 *
 *  DATE:             DECEMBER, 1984
 */

/*  THIS PROGRAM ALLOWS A USER TO MANUALLY CREATE THE
    STANDARD REPORT DEFINITIONS.  IT PROMPTS THE USER
    FOR THE REQUIRED INFORMATION AND THEN PROPERLY
    ARRANGES IT IN THE DATA FILE SRDEF.  THE OTHER
    OPTION IS TO CALL THE DATA DICTIONARY ACCESS
    PROCEDURE.  THIS LATER METHOD AUTOMATICALLY
    CREATES THE SRDEF FILE.
 */

        DCL SRDEF FILE RECORD OUTPUT SEQUENTIAL ENV(F(1950));
        DCL 1 REPORT,
              2 RID     FIXED(2),       /* REPORT ID */
              2 FNAME   CHAR(32),       /* REPORT FULL NAME */
              2 DES     CHAR(72),       /* REPORT DESCRIPTION */
              2 T_NUM   FIXED(2),       /* NO. OF TARGET ITEMS */
              2 Q_NUM   FIXED(2),       /* NO. OF QUALIFIERS */
              2 TITEM(10),
                3 TINAME CHAR(12),      /* ITEM NAME */
                3 FORMAT CHAR(10),      /* FORMAT */
                3 DBNAME CHAR(8),       /* DATABASE NAME */
                3 LOGID  CHAR(8),       /* LOGIN ID */
                3 PASSWD CHAR(8),       /* PASSWORD */
                3 FULNAM CHAR(32),      /* FULL NAME OF THE ITEM */
                3 TIDES  CHAR(72),      /* DESCRIPTION OF THE ITEM */
                3 HEIGHT FIXED(2),      /* LENGTH OF ACCESS PATH */
                3 PATH(4) CHAR(8);      /* ARRAY OF PARENT SEGMENTS */
        DCL (I,J,K)  FIXED(2);
        DCL YESORNO CHAR(3);
        DCL NUMBER  FIXED(2);


/*********************/
/*  MAIN PROCEDURE  */
/*********************/

        OPEN FILE (SRDEF);
        DO I=1 TO 3;                    /* INPUT REPORT RECORD */
         DISPLAY('NOW ENTER REPORT RECORD');
         RID = I;
         DO J=1 TO 10;                  /* INIT WORK AERA */
            TITEM(J).TINAME = '';
            TITEM(J).FORMAT = '';
            TITEM(J).DBNAME = '';
            TITEM(J).LOGID  = '';
            TITEM(J).PASSWD = '';
            TITEM(J).FULNAM = '';
            TITEM(J).TIDES  = '';
            TITEM(J).HEIGHT = '';
            DO K=1 TO 4;
               TITEM(J).PATH(K) = '';
            END;
         END;
         DISPLAY('ENTER REPORT FULL NAME--CHAR(32)')
           REPLY(FNAME);
         DISPLAY('ENTER DES OF REPORT---CHAR(72)')
           REPLY(DES);
         DISPLAY('ENTER NO. OF TARGET ITEMS');
         GET LIST (NUMBER);
         T_NUM = NUMBER;
         DO J=1 TO T_NUM;               /* TARGET ITEM */
            CALL GETITEM(J);
        END;
        DISPLAY('ENTER NO. OF QUALIFIERS');
        GET LIST (NUMBER);
        Q_NUM = NUMBER;
        DO J=1 TO Q_NUM;                /* QUALIFIER */
```

```
            CALL GETITEM(J + T_NUM);
          END;

          WRITE FILE (SRDEF) FROM (REPORT);
          DISPLAY('WRITE REPORT RECORD ONCE');
          DISPLAY('ANY MORE REPORTS?  (YES OR NO)') REPLY(YESORNO);
          IF YESORNO NE 'YES' THEN GO TO ZSREND;
        END; /* REPORT */


/*****************************************************************/
/*                                                               */
/*  PROCEDURE    GETITEM                                         */
/*                                                               */
/*  PURPOSE:   THIS PROCEDURE PROMPTS THE USER TO ENTER          */
/*             A DATA ITEM.                                      */
/*                                                               */
/*****************************************************************/

GETITEM: PROCEDURE(P);              /* ENTER ITEM DATA */
         DCL P  FIXED(2);
          DISPLAY('NOW ENTER ITEM');
          DISPLAY('ENTER ITEM NAME---CHAR(12)')
            REPLY(TITEM(P).TINAME);
          DISPLAY('ENTER FORMAT--CHAR(10)')
            REPLY(TITEM(P).FORMAT);
          DISPLAY('ENTER DATABASE NAME---CHAR(8)')
            REPLY(TITEM(P).DBNAME);
          DISPLAY('ENTER LOGIN ID---CHAR(8)')
            REPLY(TITEM(P).LOGID);
          DISPLAY('ENTER PASSWORD---CHAR(8)')
            REPLY(TITEM(P).PASSWD);
          DISPLAY('ENTER FULL NAME OF THE ITEM---CHAR(32)')
            REPLY(TITEM(P).FULNAM);
          DISPLAY('ENTER DESCRIPTION OF THE ITEM---CHAR(72)')
            REPLY(TITEM(P).TIDES);
          DISPLAY('ENTER LENGTH OF ACCESS PATH');
          GET LIST (NUMBER);
          TITEM(P).HEIGHT = NUMBER;
          DISPLAY('NOW ENTER THE ACCESS PATH');
          DO K=1 TO TITEM(P).HEIGHT;  /* ACCESS PATH */
             DISPLAY('ENTER PARENT SEGMENT NAME---CHAR(8)')
               REPLY(TITEM(P).PATH(K));
          END; /* ACCESS PATH */
        END; /* ITEM */

       CLOSE FILE (SRDEF);
ZSREND:DISPLAY('SRDEF FILE COMPLETED');
        END ZSRDEF;
```

```
YUSINT:
    PROCEDURE OPTIONS(MAIN);

/*
 *   PROJECT:          REPORT GENERATOR
 *   PROGRAMMER:       DEPARTMENT OF COMPUTER SCIENCE
 *                     UNIVERSITY OF ILLINOIS
 *
 *   FILENAME:         FRONTENDDRIVER.PLI
 *   LANGUAGE:         PL/I UNDER CMS
 *   DEPENDENCIES:     INITTHESAURUS.PLI          INITREPORTDFNS.PLI
 *                     SETDBINTERFACE.PLI         QUERYPARSER.PLI
 *                     DECISIONMAKER.PLI
 *
 *   DATE:            DECEMBER, 1984
 */


/*   THIS MODULE CONTAINS THE DRIVER FOR THE FRONT END
     OF THE REPORT GENERATOR.
*/


  DCL SYSIN FILE INPUT STREAM ENV ( F RECSIZE (80));
  DCL URIFR FILE OUTPUT STREAM ENV(F(236));
  DCL SYSPRINT FILE OUTPUT STREAM ENV ( F RECSIZE (80));

  DCL YQINIT ENTRY EXTERNAL,      /* INITIALIZATION OF THESAURUS */
      YUINIT ENTRY EXTERNAL,      /* INITIALIZATION OF STANDARD REPORT */
                                  /* DEFINITIONS */
      YAINIT ENTRY EXTERNAL,      /* SET UP DATA DICTIONARY INTERFACE */
      YQPARSE ENTRY EXTERNAL,     /* QUERY PARSER */
      ZUIF ENTRY EXTERNAL;        /* DECISION MAKER */

  DCL YES BIT(1) INIT('1'),
      NO BIT(1) INIT('0');
  DCL MORE_Q BIT(1) INIT('1') STATIC EXTERNAL;   /* ANOTHER QUERY */


/*********************/
/*   MAIN PROCEDURE  */
/*********************/

  CALL YQINIT;
  CALL YUINIT;
  CALL YAINIT;
  OPEN FILE(URIFR);
   PUT FILE(URIFR) EDIT(-1) (COL(1),F(2));
   CLOSE FILE(URIFR);
     CALL YQPARSE;
     IF MORE_Q THEN CALL ZUIF;
  END YUSINT;
```

```
  YQINIT:
     PROCEDURE OPTIONS(MAIN);

/*
 *   PROJECT:          REPORT GENERATOR
 *   PROGRAMMER:       DEPARTMENT OF COMPUTER SCIENCE
 *                     UNIVERSITY OF ILLINOIS
 *
 *   FILENAME:         INITTHESAURUS.PLI
 *   LANGUAGE:         PL/I UNDER CMS
 *   DEPENDENCIES:     SKWDOUT.DATA
 *
 *   DATE:             DECEMBER, 1984
 */

/*   THIS MODULE INITIALIZES THE PARSER'S INTERNAL REPRESENTATION
     OF THE THESAURUS.   IT SIMPLY READS THE DATA FROM THE FILE
     SKWDOUT.DATA AND CREATES THE HASH TABLE.
*/

   DCL SKWDOUT FILE INPUT STREAM ENV(F RECSIZE(80));
   DCL NULL BUILTIN;
   DCL YES BIT(1) INIT('1'B),
       NO BIT(1) INIT('0'B),
       PTR POINTER;
   DCL HTBSIZE DEC FIXED INIT(1213),       /* HASH TABLE SIZE */
       TBSIZE DEC FIXED INIT(1013),        /* OVERFLOW TABLE SIZE */
       BKSIZE DEC FIXED INIT(4);           /* BUCKET SIZE */
   DCL 1 HTABLE(0:1212) STATIC EXTERNAL,   /* HASH TABLE--THESAURUS */
         2 TBL(4),
           3 SKWDF CHAR(12),               /* USER SEARCH KEYWORD */
           3 COUNT DEC FIXED INIT((1213*4)0), /*NUMBER INTERNAL KEYWORDS*/
           3 INKWDPT POINTER,              /* NEXT INTERNAL KEYWORD */
         2 OVERFL DEC FIXED INIT((1013)0);
   DCL 1 INKWDREC BASED(CURR_INKWD_PT),    /* NODE OF INTERNAL KEYWORD LIST */
         2 INKWDF CHAR(12) VARYING,        /* INTERNAL KEYWORD */
         2 NEXTINKWD POINTER;              /* NEXT INTERNAL KEYWORD */

   DCL (I,J) DEC FIXED,
       MORE_REC BIT(1) INIT('1'B);


/*********************/
/*  MAIN PROCEDURE  */
/*********************/

   ON ENDFILE(SKWDOUT) MORE_REC=NO;
   DO WHILE (MORE_REC);
   GET FILE(SKWDOUT) EDIT(I,J) (COL(1),F(5),X(1),F(5));
   IF J=0 THEN
      GET FILE(SKWDOUT) EDIT (OVERFL(I)) (X(1),F(5));
   ELSE DO;
      GET FILE(SKWDOUT)EDIT(SKWDF(I,J),COUNT(I,J))(X(1),A(12),X(1),F(5));
      PTR=NULL;
      GET FILE(SKWDOUT) SKIP;
      DO K=1 TO COUNT(I,J);
         ALLOCATE INKWDREC;
         GET FILE(SKWDOUT) (INKWDF);
         NEXTINKWD=PTR;
         PTR=CURR_INKWD_PT;
      END; /* END DO K */
      INKWDPT(I,J)=PTR;
   END, /*END ELSE */
   END, /*END WHILE*/
 END YQINIT;
```

```
YQPARSE:
   PROCEDURE OPTIONS(MAIN).

/*
 *  PROJECT          REPORT GENERATOR
 *  PROGRAMMER:      DEPARTMENT OF COMPUTER SCIENCE
 *                   UNIVERSITY OF ILLINOIS
 *
 *  FILENAME:        QUERYPARSER.PLI
 *  LANGUAGE:        PL/I UNDER CMS
 *  DEPENDENCIES:    ACCESSDICTIONARY.PLI
 *
 *  DATE:            DECEMBER, 1984
 */

/* THIS MODULE IMPLEMENTS THE QUERY PARSING.
   IT ANALYZES THE USER'S QUERY AND EXTRACTS THE
   TARGET ITEMS AND/OR THE QUALIFYING ITEMS FROM THE QUERY.
     (FOR DETAILED INFORMATION SEE THE THESAURUS
      OF USER SEARCH KEYWORDS (TSK) AND STATE
      DIAGRAM DESCRIBED IN THE EXTERNAL DOCUMENTATION.)
   THEN THE DATA DICTIONARY IS ACCESSED TO OBTAIN
   THE ACTUAL DATABASE NAMES CORRESPONDING TO THESE
   ITEMS WHICH ARE PASSED TO THE DECISION MAKER. */


  DCL NULL BUILTIN;
  DCL (SKWD,INKWD) CHAR(12),                  /* USER SEARCH KEYWORD*/
                                              /* INTERNAL KEYWORD */
      BUFSZ DEC FIXED(4) INIT(800),           /* BUFFER SIZE */
      BUF(BUFSZ) CHAR(1),                     /* INPUT QUERY BUFFER */
      YES BIT(1) INIT('1'B),
      NO BIT(1) INIT('0'B),
      MORE_Q BIT(1) STATIC EXTERNAL,          /* MORE LINES IN QUERY */
      MORE_WD BIT(1),                         /* MORE WORDS IN LINE */
      WD CHAR(12) VARYING,                    /* TOKEN OR WORD */
      STATE DEC FIXED(1),                     /* STATE OF SYSTEM */
      FOUND BIT(1),
      QCON BIT(1) INIT(YES),                  /* CONTINUE QUERYING */
      (WPOS,WEND,ITEMPOS,ITEMEND) DEC FIXED(3),
         /* WORD POSITION; WORD END; ITEM POSITION; ITEM END */
      QLENG DEC FIXED(3),                     /* LENGTH OF QUERY */
      ERR DEC FIXED(1),                       /* ERROR FLAG */
      YESORNO CHAR (4),
      PTR POINTER;
  DCL HTBSIZE DEC FIXED INIT(1213),           /* HASH TABLE SIZE */
      TBSIZE DEC FIXED INIT(1013),            /* OVERFLOW TABLE SIZE */
      BKSIZE FIXED INIT(4);                   /* BUCKET SIZE */
  DCL 1 HTABLE(0:1212) STATIC EXTERNAL,/* HASH TABLE--THESAURUS */
        2 TBL(4),
          3 SKWDF CHAR(12),                   /* USER SEARCH KEYWORD */
          3 COUNT DEC FIXED INIT((1213*4)0),
          3 INKWDPT POINTER,                  /* TO INTERNAL KEYWORD LIST */
        2 OVERFL DEC FIXED INIT((1013)0),
  DCL 1 YSKWDL BASED(CURR_SKWD_PT),           /* USER SEARCH KEYWORD */
        2 IKCOUNT DEC FIXED(2),               /* NUMBER OF ASSOCIATED */
                                              /* INTERNAL KEYWORDS */
        2 IKPTR POINTER,                      /* NEXT INTERNAL KEYWORD */
        2 NEXTSK POINTER;                /* NEXT USER SEARCH KEYWORD */
  DCL 1 INKWDREC BASED(CURR_INKWD_PT),  /* INTERNAL KEYWORD RECORD*/
        2 INKWDF CHAR(12) VARYING,            /* INTERNAL KEYWORD NAME */
        2 NEXTINKWD POINTER,                  /* NEXT INTERNAL KEYWORD */
  DCL 1 YTARGET BASED(CURR_TARGET_PT), /* NODE OF TARGET ITEM LIST*/
          2 STCOUNT DEC FIXED(2),             /* NUMBER OF ASSOCIATED */
                                              /* INTERNAL KEYWORDS */
          2 MATCH BIT(1) INIT('1'B),
          2 STQPTR POINTER,                   /* TO INTERNAL KEYWORD LIST */
          2 NEXTTG POINTER;                   /* NEXT TARGET ITEM */
  DCL 1 YQUALI BASED(CURR_QUALI_PT),          /* NODE OF QUALIFIER LIST */
        2 SQCOUNT DEC FIXED(2),               /* NUMBER OF ASSOCIATED */
                                              /* INTERNAL KEYWORDS */
        2 QVALUEU CHAR(12),                   /* QUALIFIER VALUE */
        2 QVALUEB CHAR(12),
        2 MATCH BIT(1) INIT('1'B),
        2 SQUPTR POINTER,                     /* TO INTERNAL KEYWORD LIST */
        2 NEXTQU POINTER;                     /* NEXT QUALIFIER ITEM */
  DCL (TGHEAD,QUHEAD) POINTER STATIC EXTERNAL,
                  /* HEAD OF TARGET AND QUALIFIER ITEM LIST */
  DCL (TCOUNT,QCOUNT) DEC FIXED(2),           /* NUMBER TARGET ITEMS */
```

```
                                        /* NUMBER QUALIFIER ITEMS */
  DCL SKHEAD POINTER;       /* HEAD OF USER SEARCH KEYWORD LIST */
  DCL HEAD POINTER STATIC EXTERNAL;
                  /* HEAD TO LIST OF ACTUAL DATABASE ITEM NAMES */
  DCL CATPTR POINTER STATIC EXTERNAL;   /* CATAGORY PTR */
  DCL TMEM CHAR(40) STATIC EXTERNAL;
                  /* TARGET MEMBER -- SINGLE KEYWORD FROM FORMAT */
  DCL 1 CATREC BASED(CATPTR),           /* NODE OF CATAGORY LIST */
        2 CATMEMN CHAR(32),             /* CATAGORY MEMBER NAME */
        2 CATMEMT CHAR(32),             /* CATAGORY ITEM NAME */
        2 CATNEXT POINTER;              /* NEXT CATAGORY */
    DCL ZACSD3 ENTRY EXTERNAL;        /* DATA DICTIONARY MODULE */
    DCL FORMLENG DEC FIXED(2) INIT(40), /* FORMAT LENGTH */
        FORMWD(50) CHAR(FORMLENG) VARYING INIT((50)'');
                  /* KEYWORD FORMAT FOR DICTIONARY ACCESS */
  DCL NOTG DEC FIXED(1) INIT(1),        /* NO TARGET ITEM */
      NOTITEM DEC FIXED(1) INIT(2),
                                /* TARGET ITEM HAS NO DATA ITEM */
      NOQITEM DEC FIXED(1) INIT(3),
                                /* QUALIFIER ITEM HAS NO DATA ITEM */
      MISQUP DEC FIXED(1) INIT(4),
                                /* QUALIFYING ITEM PART MISSING */
      MISQUV DEC FIXED(1) INIT(5), /* QUALIFIER VALUE MISSING */
      ILGCHR DEC FIXED(1) INIT(6);   /* ILLEGAL CHARACTER */


/*********************/
/*  MAIN PROCEDURE  */
/*********************/

  CALL INTRO;

  DO WHILE(QCON),
      CALL INITQ;
      CALL NEWTG;
      MORE_Q=QUERYIN;
      MORE_WD=MORE_Q;
      QCON=MORE_Q;
      STATE=0;
      DO WHILE(MORE_WD);
          WD=GETWD;
          IF WD='' THEN MORE_WD=NO;
          ELSE DO;
              SELECT(STATE);
                  WHEN(0) DO;
                   ITEMEND=WEND;
                    SELECT;
                     WHEN(CKWD(WD)) DO;
                         STATE=2;
                         ITEMPOS=WPOS;
                      END;
                     WHEN(EQU(WD));
                     WHEN(CONJ(WD));
                     OTHERWISE DO;
                         ITEMPOS=WPOS;
                         STATE=1;
                     END;
                   END; /* END SELECT */
                  END; /* END 0 */
                  WHEN(1) DO;
                      ITEMEND=WEND;
                              IF (CKWD(WD)) THEN DO;
                                  STATE=2;
                                  ITEMPOS=WPOS;
                              END;
                              ELSE IF (CONJ(WD) OR QUALIFIER(WD)) THEN DO;
                                  CALL MESG(NOTITEM,'',ITEMPOS,ITEMEND);
                                  IF (QUALIFIER(WD)) THEN DO;
                                      STATE=3;
                                      CALL NEWQU;
                                  END;
                          END;
                          END;
                  WHEN(2) DO;
                              SELECT;
                               WHEN(CONJ(WD)) DO;
                                     STATE=0;
                                     TGHEAD->STGPTR=SKHEAD;
                                     CALL ACSDD(1);
                                     CALL NEWTG;
                               END;
```

```
                    WHEN(QUALIFIER(WD)) DO;
                          STATE=3;
                          TGHEAD->STGPTR=SKHEAD;
                          CALL ACSDD(1);
                          CALL NEWQU;
                    END;
                    OTHERWISE DO;
                        ITEMEND=WEND;
                        IF (CKWD(WD)) THEN DO;
                          TGHEAD->STCOUNT=TGHEAD->STCOUNT+1;
                        END;
                     END;
              END;
            END;
      WHEN(3) DO;
                IF (TCOUNT=0) THEN DO;
                  CALL MESG(NOTG,'',0,0);
                END;
                ELSE
                SELECT;
                 WHEN(CONJ(WD));
                 WHEN(EQU(WD));
                 WHEN(CKWD(WD)) DO;
                     STATE=5;
                     ITEMPOS=WPOS;
                     ITEMEND=WEND;
                 END;
                 OTHERWISE DO;
                     STATE=4;
                     ITEMPOS=WPOS;
                     ITEMEND=WEND;
                 END;
                END; /* END SELECT */
              END; /* END 3 */
    WHEN(4) DO;
          SELECT;
           WHEN(EQU(WD)) CALL MESG(NOQITEM,WD,ITEMPOS,ITEMEND);
           WHEN(CONJ(WD)) CALL MESG(MISQUP,'',ITEMPOS,ITEMEND);
           OTHERWISE DO;
               IF (CKWD(WD)) THEN DO;
                 STATE=5;
               END;
               ITEMEND=WEND;
          END; /* END OTHERWISE*/
       END; /* END SELECT */
            END; /* END WHEN(4) */
      WHEN(5) DO;
                IF EQL(WD) THEN DO;
                  STATE=6;
                  QUHEAD->SQUPTR=SKHEAD;
                  CALL ACSDD(2);
                END;
                ELSE IF (CONJ(WD)) THEN
                  CALL MESG(MISQUV,'',ITEMPOS,ITEMEND);
                ELSE DO;
                  ITEMEND=WEND;
                  IF (CKWD(WD)) THEN DO;
                     QUHEAD->SQCOUNT=QUHEAD->SQCOUNT+1;
                  END;
                END; /* END ELSE */
              END; /* END WHEN(5) */
      WHEN(6) DO;
                IF (CONJ(WD)) THEN
                  CALL MESG(MISQUV,'',ITEMPOS,ITEMEND);
                ELSE IF (NOT EQL(WD)) THEN DO;
                  QUHEAD->QVALUEU=WD;
                  STATE=7;
                END;
              END; /* END WHEN(6) */
      WHEN(7) DO;
                CALL NEWQU;
                IF CONJ(WD) THEN STATE=3;
                ELSE DO;
                  ITEMPOS=WPOS;
                  IF (CKWD(WD)) THEN DO;
                    STATE=5;
                  END;
                  ELSE STATE=4;
                END; /*END ELSE */
        END; /* END 7 */
  END; /* END SELECT */
```

```
        END; /* END ELSE */
      END; /*END WHILE MORE_WD */
      IF (MORE_Q) THEN DO;
      SELECT;
      WHEN(STATE=0) DO;
          TGHEAD=TGHEAD->NEXTTG;
          TCOUNT=TCOUNT-1;
          IF (TCOUNT=0) THEN DO;
              CALL MESG(NOTG,'',0,0);
        END;
        END;
     WHEN(STATE=1) DO;
            CALL MESG(NOTITEM,'',ITEMPOS,ITEMEND);
            TGHEAD=TGHEAD->NEXTTG;
            TCOUNT=TCOUNT-1;
            IF (TCOUNT=0) THEN CALL MESG(NOTG,'',0,0);
     END;
       WHEN(STATE=2) DO;
            TGHEAD->STGPTR=SKHEAD;
            ITEMEND=WEND;
            CALL ACSDD(1);
            IF (STATE=0) THEN DO;
                TCOUNT=TCOUNT-1;
                 TGHEAD=TGHEAD->NEXTTG;
             END;
            IF (TCOUNT=0) THEN
               CALL MESG(NOTG,'',0,0);
                     END;
      WHEN(STATE=3 OR STATE=4) DO;
        IF (STATE=4) THEN CALL MESG(MISQUP,'',ITEMPOS,ITEMEND);
        QUHEAD=QUHEAD->NEXTQU;
        QCOUNT=QCOUNT-1;
         END;
       WHEN(STATE=5 OR STATE=6) DO;
            CALL MESG(MISQUV,'',ITEMPOS,ITEMEND);
            QUHEAD=QUHEAD->NEXTQU;
            QCOUNT=QCOUNT-1;
            END;
       WHEN(STATE=7);
      END; /* END SELECT */
     IF (ERR=0) THEN QCON=NO;
     ELSE IF (ERR NE 1) THEN DO;
        PUT EDIT ('DO YOU WANT TO CONTINUE THIS QUERY? ')
          (COL(1),A);
        CALL YORN (YESORNO);
        IF YESORNO = 'YES' THEN QCON=NO;
        ELSE DO;
          PUT SKIP;
          PUT SKIP;
         PUT SKIP EDIT('REPORT GENERATOR EXITED AT USER''S REQUEST')(A);
          STOP;
        END;
    END; /* END ELSE */
  END; /* END WHILE QCON */
  END; /* END IF MORE_Q*/

/*****************************************************************/
/*                                                               */
/*   PROCEDURE    INITQ                                          */
/*                                                               */
/*   PURPOSE:     THIS PROCEDURE INITIALIZES THE VARIABLES       */
/*                FOR THE QUERY PARSER                           */
/*                                                               */
/*****************************************************************/

  INITQ:
      PROCEDURE;
      QCON=YES;
      TGHEAD=NULL;
      QUHEAD=NULL;
      TCOUNT=0;
      QCOUNT=0;
      ERR=NO;
  END INITQ;


/*****************************************************************/
/*                                                               */
/*   PROCEDURE    NEWTG                                          */
/*                                                               */
```

```
/*   PURPOSE      THIS PROCEDURE ADDS A NEW TARGET ITEM          */
/*                TO THE CURRENT LIST OF TARGET ITEMS.           */
/*                                                               */
/*****************************************************************/

  NEWTG:
     PROCEDURE;

  ALLOCATE YTARGET;
  CURR_TARGET_PT->NEXTTG=TGHEAD;
  TCOUNT=TCOUNT+1;
  TGHEAD=CURR_TARGET_PT;
  TGHEAD->STCOUNT=1;
  TGHEAD->STGPTR=NULL;
  SKHEAD=NULL;
  END NEWTG;


/*****************************************************************/
/*                                                               */
/*   PROCEDURE.   NEWQU                                          */
/*                                                               */
/*   PURPOSE.     THIS PROCEDURE ADDS A NEW QUALIFIER ITEM       */
/*                TO THE CURRENT LIST OF QUALIFIER ITEMS.        */
/*                                                               */
/*****************************************************************/

  NEWQU:
     PROCEDURE;

  ALLOCATE YQUALI;
  CURR_QUALI_PT->NEXTQU=QUHEAD;
  QCOUNT=QCOUNT+1;
  QUHEAD=CURR_QUALI_PT;
  QUHEAD->SQCOUNT=1;
  QUHEAD->SQUPTR=NULL;
  SKHEAD=NULL;
  END NEWQU;


/*****************************************************************/
/*                                                               */
/*   PROCEDURE:   CONSL                                          */
/*                                                               */
/*   PURPOSE:     THIS FUNCTION CONVERTS LOWER CASE LETTERS      */
/*                TO UPPER CASE LETTERS.                         */
/*                                                               */
/*****************************************************************/

  CONSL:
     PROCEDURE(CHR) RETURNS(CHAR(1));
  DCL CHR CHAR(1);
  SELECT(CHR);
     WHEN('A') RETURN('A');
     WHEN('B') RETURN('B');
     WHEN('C') RETURN('C');
     WHEN('D') RETURN('D');
     WHEN('E') RETURN('E');
     WHEN('F') RETURN('F');
     WHEN('G') RETURN('G');
     WHEN('H') RETURN('H');
     WHEN('I') RETURN('I');
     WHEN('J') RETURN('J');
     WHEN('K') RETURN('K');
     WHEN('L') RETURN('L');
     WHEN('M') RETURN('M');
     WHEN('N') RETURN('N');
     WHEN('O') RETURN('O');
     WHEN('P') RETURN('P');
     WHEN('Q') RETURN('Q');
     WHEN('R') RETURN('R');
     WHEN('S') RETURN('S');
     WHEN('T') RETURN('T');
     WHEN('U') RETURN('U');
     WHEN('V') RETURN('V');
     WHEN('W') RETURN('W');
     WHEN('X') RETURN('X');
     WHEN('Y') RETURN('Y');
     WHEN('Z') RETURN('Z');
     OTHERWISE RETURN(CHR);
  END;
```

```
  END CONSL;

/*****************************************************************/
/*                                                               */
/*   PROCEDURE:   QUERYIN                                        */
/*                                                               */
/*   PURPOSE:     THIS FUNCTION PROMPTS THE USER TO ENTER A     */
/*                QUERY.  IT READS THE QUERY IN ONE CHARACTER   */
/*                AT A TIME CONVERTING IT TO UPPER CASE AND     */
/*                PLACING EACH INTO A BUFFER.  IT RETURNS A     */
/*                FLAG INDICATING WHETHER A QUERY WAS           */
/*                ENTERED.                                      */
/*                      '0' -- NO QUERY OR REQUEST TO EXIT      */
/*                      '1' -- A QUERY WAS ACCEPTED             */
/*                                                               */
/*****************************************************************/


  QUERYIN:
     PROCEDURE RETURNS(BIT(1));
  DCL 1 CHR,
        2 C(80) CHAR(1);
  DCL LREC CHAR(80) DEFINED CHR;
  DCL ITEMP INIT(0);

  BUF=' ';
  LREC='1';
  I=0;
  PUT SKIP;
  PUT EDIT ('ENTER THE INITIAL QUERY') (COL(1),A);
  DO WHILE(LREC NE (80)'');
     GET EDIT(LREC) (A(80));
     IF (LREC NE (80)' ') THEN
     DO;
        ITEMP=I*80;
        DO J=1 TO 80;
           IF (C(J) GE 'A' & C(J) LE 'Z') THEN
              BUF(J+ITEMP)=CONSL(C(J));
           ELSE
              BUF(J+ITEMP)=C(J);
        END; /*END DO*/
        I=I+1;
     END; /* END THEN */
  END; /* END WHILE */
  QLENG=ITEMP+80;
  IF (BUF(1) CAT BUF(2) CAT BUF(3) CAT BUF(4) = 'ENDQ' OR I = 0)
     THEN RETURN('0');
     ELSE RETURN('1');
  END QUERYIN;


/*****************************************************************/
/*                                                               */
/*   PROCEDURE:   GETWD                                          */
/*                                                               */
/*   PURPOSE:     THIS FUNCTION DETERMINES AND RETURNS THE      */
/*                NEXT WORD OR TOKEN IN THE QUERY BUFFER.       */
/*                IF NO 'NEXT' WORD EXISTS THE NULL STRING      */
/*                IS RETURNED.                                  */
/*                                                               */
/*****************************************************************/


  GETWD:
     PROCEDURE RETURNS(CHAR(12) VARYING);
  DCL FLAG BIT(1) INIT(YES),
      WD CHAR(12) VARYING INIT(''),
      WTEMP CHAR(12) VARYING INIT(''),
      I DEC FIXED(3) STATIC INIT(1),
      WSTATE DEC FIXED(1) STATIC INIT(1),
      ILN DEC FIXED(1);
  DO WHILE(FLAG);
     SELECT(WSTATE),
      WHEN(1) DO;
                DO WHILE (I LE BUFSZ AND DELM(BUF(I)));
                   I=I+1;
                END;
                IF (I GT BUFSZ) THEN DO;
                   I=1;
                   FLAG=NO;
```

```
            END;
            ELSE
             SELECT,
                WHEN (CHRINWD(BUF(I))) WSTATE=2;
                WHEN (EQU(BUF(I))) WSTATE=3;
                WHEN (COMMA(BUF(I))) WSTATE=4;
                OTHERWISE WSTATE=5;
             END; /* END SELECT */
           END; /*END WHEN(1)*/
WHEN(2) DO;
            FLAG=NO;
            WPOS=I;
           DO WHILE(I LE BUFSZ AND CHRINWD(BUF(I)));
              WD=WD CAT BUF(I);
              I=I+1;
           END;
           IF (I LE BUFSZ) THEN WEND=I-1;
              ELSE WEND=I;
           IF (I GT BUFSZ) THEN DO;
              I=1;
              WSTATE=1;
           END; /* END IF*/
           ELSE
            SELECT;
              WHEN (DELM(BUF(I))) WSTATE=1;
              WHEN (EQU(BUF(I))) WSTATE=3;
              WHEN (COMMA(BUF(I))) WSTATE=4;
              OTHERWISE WSTATE=5;
            END; /* END SELECT */
          END;/*WHEN(2)*/
WHEN(3) DO;
           FLAG=NO;
           WD=WD CAT BUF(I);
           I=I+1;
           IF (I GT BUFSZ) THEN DO;
              I=1;
              WSTATE=1;
           END;
           ELSE
            SELECT;
           WHEN (DELM(BUF(I))) WSTATE=1;
           WHEN (CHRINWD(BUF(I))) WSTATE=2;
          WHEN (EQU(BUF(I))) WTATE=3;
            WHEN (COMMA(BUF(I))) WSTATE=4;
            OTHERWISE WSTATE=5;

        END; /* END SELECT */
      END; /* WHEN(3)*/
WHEN(4) DO;
           FLAG=NO;
           WD=WD CAT BUF(I);
           I=I+1;
           IF (I GT BUFSZ) THEN DO;
              I=1;
              WSTATE=1;
           END;
           ELSE
             SELECT,
           WHEN (DELM(BUF(I))) WSTATE=1;
           WHEN (CHRINWD(BUF(I))) WSTATE=2;
            WHEN (EQU(BUF(I))) WSTATE=3;
           WHEN(COMMA(BUF(I))) WSTATE=4;
            OTHERWISE WSTATE=5;
        END, /* END SELECT */
      END; /* WHEN(4)*/
 WHEN(5) DO;
           DO WHILE(I LE BUFSZ AND NOT LEGALCHR(BUF(I)));
              IF (STATE =5) THEN DO;
                 WTEMP=BUF(I);
                 CALL MESG(ILGCHR,WTEMP,I,I+1);
              END;
              I=I+1;
           END;
           IF (I GT BUFSZ) THEN DO;
              WSTATE=1;
              I=1;
           END; /* END IF */
           ELSE
            SELECT;
           WHEN (DELM(BUF(I))) WSTATE=1;
           WHEN (CHRINWD(BUF(I))) WSTATE=2;
```

```
                WHEN (EQU(BUF(I))) WSTATE=3;
                WHEN (COMMA(BUF(I))) WSTATE=4;
                OTHERWISE;
              END;
          END; /* END SELECT 5 */
      END; /* END SELECT */
  END; /* END WHILE */
  RETURN(WD);
  END GETWD;

/*****************************************************************/
/*                                                               */
/*   PROCEDURE:   CHRINWD                                        */
/*                                                               */
/*   PURPOSE:     THIS FUNCTION CHECKS IF A GIVEN CHARACTER      */
/*                IS A LEGAL ALPHA-NUMERIC CHARACTER OR          */
/*                AN '_'.                                        */
/*                                                               */
/*****************************************************************/

  CHRINWD:
  PROCEDURE(CHR) RETURNS(BIT(1));
  DCL CHR CHAR(1);
  RETURN(CHR='_' OR (CHR GE 'A' AND CHR LE 'Z') OR (CHR GE '0' AND
         CHR LE '9'));
  END CHRINWD;

/*****************************************************************/
/*                                                               */
/*   PROCEDURE:   DELM                                           */
/*                                                               */
/*   PURPOSE:     THIS FUNCTION CHECKS IF A GIVEN CHARACTER      */
/*                IS A DELIMITER, I.E. A BLANK.                  */
/*                                                               */
/*****************************************************************/

  DELM:
  PROCEDURE(CHR) RETURNS(BIT(1));
  DCL CHR CHAR(1);
  RETURN(CHR=' ');
  END DELM;

/*****************************************************************/
/*                                                               */
/*   PROCEDURE:   EQU                                            */
/*                                                               */
/*   PURPOSE:     THIS FUNCTION CHECKS IF A GIVEN CHARACTER      */
/*                IS AN EQUAL SIGN, '='.                         */
/*                                                               */
/*****************************************************************/

  EQU:
  PROCEDURE(CHR) RETURNS(BIT(1));
  DCL CHR CHAR(1);
  RETURN(CHR='=');
  END EQU;

/*****************************************************************/
/*                                                               */
/*   PROCEDURE:   COMMA                                          */
/*                                                               */
/*   PURPOSE:     THIS FUNCTION CHECKS IF A GIVEN CHARACTER      */
/*                IS A COMMA, ','.                               */
/*                                                               */
/*****************************************************************/

  COMMA:
  PROCEDURE(CHR) RETURNS(BIT(1));
  DCL CHR CHAR(1);
  RETURN(CHR=',');
  END COMMA;
```

```
/*****************************************************************/
/*                                                               */
/*   PROCEDURE    LEGALCHR                                       */
/*                                                               */
/*   PURPOSE      THIS FUNCTION CHECKS IF A GIVEN CHARACTER      */
/*                IS AN ALPHANUMERIC OR ONE OF THE FOLLOWING:    */
/*                A ' ', A '=' OR A ','.                         */
/*                                                               */
/*****************************************************************/


  LEGALCHR:
  PROCEDURE(CHR) RETURNS(BIT(1));
  DCL CHR CHAR(1);
  RETURN(CHRINWD(CHR) OR DELM(CHR) OR EQU(CHR) OR COMMA(CHR));
  END LEGALCHR;


/*****************************************************************/
/*                                                               */
/*   PROCEDURE:   CKWD                                           */
/*                                                               */
/*   PURPOSE:     THIS FUNCTION SEARCHES THE DATA DICTIONARY,     */
/*                A HASH TABLE, FOR THE SPECIFIED WORD.  IT      */
/*                RETURNS A FLAG INDICATING ITS PRESENCE.        */
/*                   'YES' OR '1' -- WORD FOUND                  */
/*                   'NO'  OR '0' -- NOT IN TABLE                */
/*                                                               */
/*****************************************************************/


  CKWD:
     PROCEDURE(STR) RETURNS(BIT(1));
  DCL STR CHAR(12) VARYING,
      SKWD CHAR(12),
      TMPTR POINTER,
      FLAG BIT(1) INIT('1'B);

  INDX=MOD(HASHF(STR),1013);
  SKWD=STR;
  DO WHILE(FLAG);
     I=1;
     DO WHILE(I LE 4);
           IF (COUNT(INDX,I)=0) THEN RETURN(NO);
           ELSE DO;
              IF (SKWDF(INDX,I)=SKWD) THEN DO;
                           ALLOCATE YSKWDL;
                           CURR_SKWD_PT->NEXTSK=SKHEAD;
                           SKHEAD=CURR_SKWD_PT;
                           SKHEAD->IKPTR=INKWDPT(INDX,I);
                           SKHEAD->IKCOUNT=COUNT(INDX,I);
                 RETURN(YES);
              END; /* END IF THEN */
              ELSE I=I+1;
           END; /* END ELSE IF COUNT=NE '0'B */
     END; /* END WHILE I LE 4 */
     IF (OVERFL(INDX)=0) THEN RETURN(NO);
     ELSE INDX=OVERFL(INDX);
  END; /*END WHILE FLAG */
  END CKWD;


/*****************************************************************/
/*                                                               */
/*   PROCEDURE:   CONV                                           */
/*                                                               */
/*   PURPOSE      THIS FUNCTION CONVERTS A CHARACTER TO          */
/*                ITS INTERNAL NUMERIC REPRESENTATION            */
/*                                                               */
/*****************************************************************/

  CONV:
    PROCEDURE(CHR) RETURNS(DEC FIXED(3,0));
  DCL CHR CHAR(1);
  RETURN(UNSPEC(CHR));
  END CONV;


/*****************************************************************/
/*                                                               */
/*   PROCEDURE    HASHF                                          */
```

```
/*                                                                  */
/*   PURPOSE:     THIS FUNCTION CALCULATES AND RETURNS THE          */
/*                INDEX INTO THE HASH TABLE FOR THE                 */
/*                SPECIFIED WORD.  (THE DETAILS OF THE              */
/*                HASHING FUNCTION USED CAN BE FOUND IN             */
/*                THE EXTERNAL DOCUMENTATION  )                     */
/*                                                                  */
/********************************************************************/

 HASHF:
  PROCEDURE(STR) RETURNS (DEC FIXED(12));
  DCL IND DEC FIXED(3,0),
      STR CHAR(12) VARYING,
      POS DEC FIXED(2) INIT(7) STATIC ,
      LNG DEC FIXED(2,0);
  LNG=LENGTH(STR);
  IF (LNG=0) THEN
     RETURN(0);
  ELSE
     RETURN(HASHF(SUBSTR(STR,1,LNG-1))*POS+CONV(SUBSTR(STR,LNG,1)));
  END HASHF;

/********************************************************************/
/*                                                                  */
/*   PROCEDURE    QUALIFIER                                         */
/*                                                                  */
/*   PURPOSE:     THIS FUNCTION DETERMINES IF THE SPECIFIED         */
/*                WORD IS ONE OF THE RESERVED WORDS                 */
/*                'WITH', 'WHERE', OR 'FOR'.                        */
/*                                                                  */
/********************************************************************/

  QUALIFIER:
     PROCEDURE(STR) RETURNS(BIT(1));
  DCL STR CHAR(12) VARYING;

  RETURN(STR='WITH' OR STR='WHERE' OR STR='FOR');
  END QUALIFIER;

/********************************************************************/
/*                                                                  */
/*   PROCEDURE:   EQL                                               */
/*                                                                  */
/*   PURPOSE:     THIS FUNCTION DETERMINES IF THE SPECIFIED         */
/*                WORD IS: '=', 'IS' OR 'ARE'.                      */
/*                                                                  */
/********************************************************************/

  EQL:
     PROCEDURE(STR) RETURNS(BIT(1));
  DCL STR CHAR(12) VARYING;
  RETURN(STR='=' OR STR='IS' OR STR='ARE');
  END EQL;

/********************************************************************/
/*                                                                  */
/*   PROCEDURE:   CONJ                                              */
/*                                                                  */
/*   PURPOSE:     THIS FUNCTION DETERMINES IF THE SPECIFIED         */
/*                WORD IS: 'AND'.                                   */
/*                                                                  */
/********************************************************************/

  CONJ:
     PROCEDURE(STR) RETURNS(BIT(1));
  DCL STR CHAR(12) VARYING;
  RETURN(STR='AND' OR STR=',');
  END CONJ;

/********************************************************************/
/*                                                                  */
/*   PROCEDURE    YORN                                              */
/*                                                                  */
/*   PURPOSE      THIS PROCEDURE PROMPTS THE USER TO INPUT          */
/*                'YES', 'NO' OR 'QUIT'.                            */
/*                                                                  */
/********************************************************************/
```

```
YORN:   PROCEDURE (YN);
   DCL YN CHAR(4),
   YN = '';

   DO WHILE (((YN NE 'YES') AND (YN NE 'NO')) AND (YN NE 'QUIT'));
      DISPLAY ('TYPE YES/NO')  REPLY(YN);
   END; /*DO*/
   IF YN = 'QUIT' THEN
     BEGIN;
       PUT SKIP;
       PUT SKIP;
       PUT SKIP EDIT ('REPORT GENERATOR EXITED AT USER''S REQUEST')
                     (A);
       STOP;
     END;
END YORN;


/*****************************************************************/
/*                                                               */
/*   PROCEDURE:   MESG                                           */
/*                                                               */
/*   PURPOSE:     THIS PROCEDURE DISPLAYS THE APPROPRIATE        */
/*                ERROR OR WARNING MESSAGE ON THE USER'S         */
/*                TERMINAL SCREEN.                               */
/*                                                               */
/*****************************************************************/

 MESG:
    PROCEDURE(ERRCODE,WORD,WBEGIN,WEND);
DCL ERRCODE DEC FIXED(1),
    WTMP CHAR(20) VARYING,
    WORD CHAR(12) VARYING,
    (WBEGIN,WEND,IBEGIN,IEND) DEC FIXED(3),
    ERRCON CHAR(20) VARYING, /* CONTENTS CONTAIN ERROR PART */
    ETYPE CHAR(9) VARYING, /* MESSAGE TYPE */
    MSG CHAR(50) VARYING, /*CONTENTS OF MESSAGE */
    FLAG BIT(1),
    IBEFORE DEC FIXED(1) INIT(6),
    IAFTER DEC FIXED(1) INIT(5);

 SELECT(ERRCODE);
    WHEN(NOTG) DO;
       ETYPE='ERROR: ';
       MSG='NO TARGET ITEM GIVEN, QUERY IGNORED';
       MORE_WD=NO;
     END;
    WHEN(NOTITEM) DO;
       ETYPE='WARNING: ';
       MSG=' DOES NOT SPECIFY A DATA ITEM, IGNORED ';
       TGHEAD=TGHEAD->NEXTTG;
       TCOUNT=TCOUNT-1;
       STATE=0;
       CALL NEWTG;
    END;
    WHEN(NOQITEM) DO;
       ETYPE='WARNING: ';
       MSG=' DOES NOT SPECIFY A DATA ITEM, IGNORED ';
       QUHEAD=QUHEAD->NEXTQU;
       QCOUNT=QCOUNT-1;
       WTMP=WORD;
      DO WHILE(FLAG);
       WD=GETWD;
       WTMP=WTMP CAT WD;
       FLAG=NO;
       IF (EQL(WD)) THEN FLAG=YES;
       ELSE IF (WD='') THEN
          MORE_WD=NO;
     END; /* END WHILE */
       STATE=7;
     END;
    WHEN(MISQUP) DO;
       ETYPE='WARNING  ';
       MSG=' QUALIFYING ITEM PART MISSING, IGNORED ';
       QUHEAD=QUHEAD->NEXTQU;
       QCOUNT=QCOUNT-1;
       STATE=3;
       CALL NEWQU;
     END; /* WHEN(1) */
    WHEN(MISQUV) DO;
       ETYPE='WARNING  ';
```

```
        MSG=' QUALIFIER VALUE PART MISSING FOR ';
        STATE=3;
        QUHEAD=QUHEAD->NEXTQU;
        QCOUNT=QCOUNT-1;
        CALL NEWQU;
      END; /* WHEN(2) */
     WHEN(ILGCHR) DO;
        ETYPE='ERROR: ';
        MSG=' IS AN ILLEGAL CHAR., TREATED AS A BLANK';
      END; /* WHEN(3) */
  END; /* SELECT ERRCODE */
  SELECT;
    WHEN(ERRCODE=ILGCHR) DO;
     IBEGIN=WBEGIN-IBEFORE;
     IEND=WEND+IAFTER;
     IF (IBEGIN LE 0) THEN IBEGIN=1;
     IF (IEND GT QLENG) THEN IEND=QLENG;
     ERRCON='';
     DO K=IBEGIN TO IEND;
        ERRCON=ERRCON CAT BUF(K);
     END;
     PUT EDIT(ETYPE,'''',WORD,''' IN ''',ERRCON,'''',MSG)
                          (COL(1),A,COL(2),5 A,COL(2),A);
  END; /*  WHEN ILGCHR */
  WHEN (ERRCODE=NOTG)
   PUT EDIT(ETYPE,MSG) (COL(1),A,COL(2),A);
  WHEN(ERRCODE=NOTITEM OR ERRCODE=NOQITEM) DO;
     ERRCON='';
     DO K=WBEGIN TO WEND;
        ERRCON=ERRCON CAT BUF(K);
     END;
     PUT EDIT(ETYPE,'''',ERRCON,'''',MSG)
               (COL(1),A,COL(2),4 A);
      IF (ERRCODE=NOQITEM) THEN DO;
         WTMP=ERRCON CAT WTMP;
      PUT EDIT('''',WTMP,'''') (X(1),3 A);
     END;
   END; /* END NOTITEM OR NOQITEM */
   WHEN(ERRCODE=MISQUV OR ERRCODE=MISQUP) DO;
     ERRCON='';
     DO K=WBEGIN TO WEND;
        ERRCON=ERRCON CAT BUF(K);
     END;
     PUT EDIT(ETYPE,MSG,'''',ERRCON,'''')
               (COL(1),A,COL(2),4 A);
   END; /* MISQUV */
   END; /* END SELECT */
  ERR=ERRCODE;
  END MESG;


/*****************************************************************/
/*                                                               */
/*   PROCEDURE:   ACSDD                                          */
/*                                                               */
/*   PURPOSE:     THIS PROCEDURE FOR EACH TARGET OR QUALIFIER    */
/*                ITEM ACCESSES THE DATA DICTIONARY AND          */
/*                CONVERTS THE INTERNAL KEYWORDS ASSOCIATED      */
/*                WITH THE ITEM TO A LIST OF ACTUAL DATABASE     */
/*                ITEM NAMES.                                    */
/*                                                               */
/*****************************************************************/

  ACSDD:
     PROCEDURE(TGORQU);
        DCL TGORQU DEC FIXED(1);
         DCL (RL,KL,HL,P,PP)    POINTER,
             (I,J,K)            FIXED(2);

  SELECT(TGORQU);
     WHEN(1) DO;
        PP = NULL;
               WDNUM=1;
               CALL FORM(TGHEAD->STCOUNT,TGHEAD->STGPTR,WDNUM);
     IF FORMWD(1)='' THEN WDNUM=0;
               DO K=1 TO WDNUM;
                  TMEM=FORMWD(K);
                  FORMWD(K)='';
               HEAD = NULL;
               CALL ZACSD3;    /* ACCESS D/D WITH TMEM */
               IF HEAD NE NULL THEN CALL SORTL;
```

```
            END;
          TGHEAD->STGPTR = PP;
          IF (PP = NULL) THEN DO;
             CALL MESG(NOTITEM,'',ITEMPOS,ITEMEND);
             STATE=0;
           END;
    END; /* WHEN WHEN TG */

    WHEN(2) DO;
             PP = NULL;
             WDNUM=1;
             CALL FORM(QUHEAD->SQCOUNT,QUHEAD->SQUPTR,WDNUM);
             DO K=1 TO WDNUM;
                TMEM=FORMWD(K);
                FORMWD(K)='';
                HEAD = NULL;
             CALL ZACSD3;    /* ACCESS D/D WITH TMEM */
             IF HEAD NE NULL THEN CALL SORTL;
          END;
          QUHEAD->SQUPTR = PP;
          IF (PP = NULL) THEN DO;
             CALL MESG(NOQITEM,WD,ITEMPOS,ITEMEND);
           END;
    END; /* WHEN QUALIFIER */
    OTHERWISE;
    END; /* END SELECT */
         GO TO ACSD0;


/*****************************************************************/
/*                                                               */
/*   PROCEDURE:   SORTL                                          */
/*                                                               */
/*   PURPOSE:     THIS PROCEDURE CHECKS ALL WORDS IN THE LIST    */
/*                OF DATABASE ITEM NAMES RETURNED FROM THE       */
/*                DATA DICTIONARY TO SEE IF THEY ARE CURRENTLY   */
/*                IN THE LIST.  IF NOT, THEN THEY ARE ADDED TO   */
/*                THE END OF THE LIST.                           */
/*                                                               */
/*****************************************************************/

 SORTL:   PROCEDURE;
            IF PP = NULL THEN
             BEGIN;
              I = 1;
              HL = HEAD;
              P = PP;
              DO WHILE (HL NE NULL);
                 ALLOCATE INKWDREC;
                 CURR_INKWD_PT->NEXTINKWD = NULL;
                 CURR_INKWD_PT->INKWDF = HL->CATREC.CATMEMN;
                 IF I > 1 THEN P->NEXTINKWD = CURR_INKWD_PT;
                  ELSE PP = CURR_INKWD_PT;
                 HL = HL->CATREC.CATNEXT;
                 P = CURR_INKWD_PT;
                 I = I + 1;
              END;   /* HL */
            END;
           ELSE BEGIN;    /* FOR NEXT INTERNAL KEYWORD */
              HL = HEAD;
              DO WHILE (HL NE NULL);
                 P = PP;
                 DO WHILE (P NE NULL);
                    IF HL->CATREC.CATMEMN = P->INKWDF THEN
                       GO TO SO2;
                    IF P->NEXTINKWD  NE NULL   THEN
                    P = P->NEXTINKWD;
                   ELSE  GO  TO  SO1;
                  END;
 SO1:             ALLOCATE INKWDREC;
                  CURR_INKWD_PT->INKWDF = HL->CATREC.CATMEMN;
                  CURR_INKWD_PT->NEXTINKWD = NULL;
                  P->NEXTINKWD = CURR_INKWD_PT;
 SO2              HL = HL->CATREC.CATNEXT;
              END;
            END;
           END SORTL;
 ACSD0    END ACSDD;

/*****************************************************************/
```

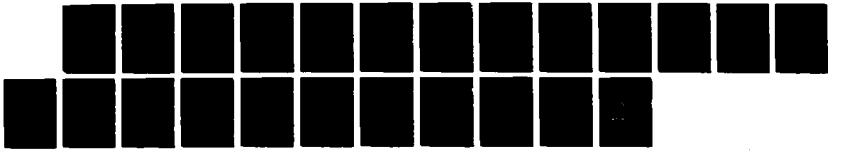
```
/*                                                                  */
/*   PROCEDURE:    FORM                                             */
/*                                                                  */
/*   PURPOSE:      THIS PROCEDURE FORMATS A CHARACTER STRING OF     */
/*                 ALL THE INTERNAL KEYWORDS ASSOCIATED WITH        */
/*                 THE SPECIFIED TARGET OR QUALIFIER ITEM.          */
/*                 EACH KEYWORD IS DELIMITED BY " AND THEY          */
/*                 ARE SEPARATED BY A ','.                          */
/*                                                                  */
/********************************************************************/

  FORM:
     PROCEDURE(SKNUM,PTR,WDNUM) RECURSIVE;
  DCL (PTR,PTM) POINTER;

  DO I=WDNUM TO 1 BY -1;
     TMP=PTR->IKCOUNT;
     PTM=PTR->IKPTR;
     DO J=TMP TO 1 BY -1;
        IF FORMWD(I)='' THEN DO;
           FORMWD((I-1)*TMP+J)='''' CAT PTM->INKWDF CAT '''';
        END;
        ELSE DO;
           FORMWD((I-1)*TMP+J)=FORMWD(I) CAT ',' CAT '''' CAT
                               PTM->INKWDF CAT '''';
        END;
        PTM=PTM->NEXTINKWD;
     END;
  END;
  WDNUM=WDNUM*TMP;
  IF (SKNUM=1) THEN DO;
     RETURN;
  END;
  ELSE DO;
     CALL FORM(SKNUM-1,PTR->NEXTSK,WDNUM);
  END;
  END FORM;


/********************************************************************/
/*                                                                  */
/*   PROCEDURE:    INTRO                                            */
/*                                                                  */
/*   PURPOSE:      THIS PROCEDURE DISPLAYS THE INTRODUCTION         */
/*                 TO THE REPORT GENERATOR ON THE USER'S            */
/*                 TERMINAL SCREEN.                                 */
/*                                                                  */
/********************************************************************/

 INTRO:  PROCEDURE;

  PUT SKIP;
  PUT SKIP EDIT ('**** WELCOME TO THE REPORT GENERATOR ****')
     (COL(15),A);
  PUT SKIP;
  PUT SKIP;
  PUT SKIP EDIT ('THIS SYSTEM IS A QUESTION - DRIVEN INTERACTIVE ONE.')
     (COL(8), A);
  PUT SKIP EDIT ('IT ACCEPTS AN INITIAL QUERY FROM THE USER, E.G.,')
      (COL(8),A);
  PUT SKIP EDIT ('SHOW THE COST FOR PROJECT = B12 AND YEAR = 1981')
     (COL(11),A);
  PUT SKIP EDIT ('AND DECIDES WHICH REPORTS SATISFY THE INITIAL QUERY')
     (COL(8),A);
  PUT SKIP EDIT ('AND THEN GENERATES THE REPORT SELECTED BY THE USER.')
     (COL(8),A);
  PUT SKIP;
  PUT SKIP EDIT ('JUST FOLLOW THE PROMPTS AND PROCEED THROUGH TO')
     (COL(8),A);
  PUT SKIP EDIT ('OBTAIN THE DESIRED REPORT.')  (COL (8),A);
  PUT SKIP;
  PUT SKIP EDIT ('NOTE   THE REPORT GENERATOR MAY BE EXITED ANY')
               (COL (8),A);
  PUT SKIP EDIT ('TIME A PROMPT IS PRESENTED BY TYPING "QUIT"')
               (COL (15), A);
  PUT SKIP EDIT (' EXCEPT WHEN PROMPTED WITH')(COL(15),A);
  PUT SKIP EDIT ('"ENTER INITIAL QUERY", THEN EITHER "ENDQ"')
               (COL (15),A);
  PUT SKIP EDIT (' OR A NULL LINE ARE REQUIRED TO EXIT ')(COL(15),A);
  PUT SKIP;
  PUT SKIP;

 END INTRO;
  END YQPARSE;
```

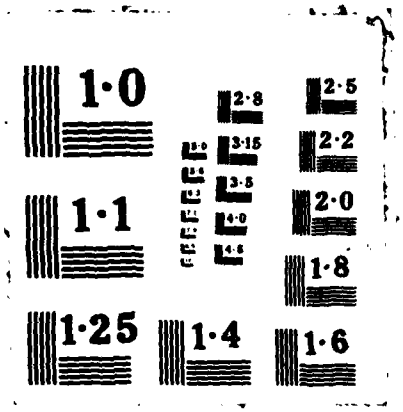
```
YAINIT   PROCEDURE OPTIONS(MAIN);

/*
 *   PROJECT:          REPORT GENERATOR
 *   PROGRAMMER:       DEPARTMENT OF COMPUTER SCIENCE
 *                     UNIVERSITY OF ILLINOIS
 *
 *   FILENAME:         SETDBINTERFACE.PLI
 *   LANGUAGE:         PL/I UNDER CMS
 *   DEPENDENCIES:     DMRUS
 *
 *   DATE:             DECEMBER, 1984
 */


/*   THIS MODULE SETS UP THE PROPER INTERFACE WITH THE DATA
     DICTIONARY SO IT CAN BE ACCESSED LATER.
*/

/* DMRUS IS THE INTERFACE PROGRAM BETWEEN USER PROGRAM AND DATA DICTIONARY
   SUPPLIED BY DATA MANAGER */
 DCL DMRUS ENTRY EXTERNAL OPTIONS(ASM,INTER);

 DCL DMOUT FILE OUTPUT STREAM ENV (F RECSIZE(80));
 DCL 1 DOUTPUT CHAR(150);

/* DCONTRO CONTAINS THE PARAMETERS NEEDED TO ACCESS DATA DICTIONARY */
 DCL 1 DCONTRO STATIC EXTERNAL,
        3 DCOMMUNE CHAR (64),
        3 DDMR     CHAR (8) INIT('    DM00'),
        3 DBUFFLEN FIXED BIN (27) ALIGNED INIT(800),
        3 DINPLEN  FIXED BIN (15) ALIGNED INIT(72),
        3 DINPLREC FIXED BIN (15) ALIGNED INIT(72),
        3 DOUTLEN  FIXED BIN (15) ALIGNED INIT(150),
        3 DOUTLREC FIXED BIN (15) ALIGNED,
        3 DFUNC    PIC '9' INIT(1),
        3 DRETURN  PIC '9',
        3 DRETURN1 PIC '9',
        3 DRETURN2 PIC '9',
        3 DSEVRITY CHAR (1),
        3 DOPTION1 PIC '9' INIT(1),
        3 DOPTION2 PIC '9' INIT(3),
        3 DOPTION3 PIC '9',
        3 DOPTION4 PIC '9',
        3 DOPTION5 PIC '9' INIT(2),
        3 DOPTION6 PIC '9' INIT(1),
        3 DOPTION7 PIC '9' INIT(0),
        3 DOPTION8 PIC '9' INIT(0),
        3 DOPTION9 PIC '9',
        3 DOUTWAIT FIXED DEC (5),
        3 DOUTTOTL FIXED DEC (5),
        3 DOUTTOTM FIXED DEC (5),
        3 DINPNO   FIXED DEC (3),
        3 FILLER00001 CHAR (3),
        3 DINPUT   CHAR (72);


/*********************/
/*   MAIN PROCEDURE  */
/*********************/

  DCOMMUNE=LOW(64),
  DINPUT = 'DICTIONARY DTEST   ',

/* THE DICTIONARY NAME HAS TO BE CHANGED TO ACCESS DIFFERENT DATA DICTIONARY*/
  CALL DMRUS(DCONTRO,DOUTPUT),
  DINPUT = 'AUTHORITY ''UIUCDCS''     ',
  CALL DMRUS(DCONTRO DOUTPUT),
 END YAINIT,
```

```
ZACSD3 : PROCEDURE OPTIONS(MAIN);

/*
 *   PROJECT:          REPORT GENERATOR
 *   PROGRAMMMER:      DEPARTMENT OF COMPUTER SCIENCE
 *                     UNIVERSITY OF ILLINOIS
 *
 *   FILENAME:         ACCESSDICTIONARY.PLI
 *   LANGUAGE:         PL/I UNDER CMS
 *   DEPENDENCIES:     DMRUS
 *
 *   DATE:             DECEMBER, 1984
 */

/*   THIS MODULE ACCESSES THE DATA DICTIONARY THROUGH THE DMRUS
     INTERFACE USING THE COMMAND 'WHAT FORMS ...' IN ORDER TO
     RETRIEVE MEMBER NAMES FOR A REPORT'S KEYWORDS IN THE
     CATEGORY CLASSIFICATION.

*/

  DCL DMRUS ENTRY EXTERNAL OPTIONS(ASM,INTER);
  DCL NULL BUILTIN;
  DCL MOD  BUILTIN;
  DCL 1 CATREC BASED(CATPTR),                    /* NODE OF CATEGORY LIST *
        2 CATMEMN CHAR(32),                      /* CATEGORY MEMBER NAME *
        2 CATMEMT CHAR(32),                      /* CATEGORY MEMBER TYPE *
        2 CATNEXT POINTER;                       /* NEXT CATEGORY */

    DCL HEAD     STATIC EXTERNAL POINTER;        /* HEAD OF CATEGORY LIST *
    DCL CATPTR   STATIC EXTERNAL POINTER;        /* CATEGORY PTR */
    DCL TMEM     STATIC EXTERNAL CHAR(40);       /* TARGET MEMBER */
    DCL TOTALN   STATIC EXTERNAL CHAR(5);

 DCL
 1 DPQB_BASED4                                   BASED (DMR_PTR),
   3 FILLER00010                                 CHAR (1),
   3 DPQBDTLL                                    /*DETAIL LINE                *
     5 DMTEST                                      CHAR(4),
     5 FILLER00011 (10)                          CHAR (1),
     5 DPQBDMTP                                  CHAR (32)
                                                 /*MEMBER-TYPE                *
     5 FILLER00012                               CHAR (1),
     5 DPQBDMEM                                  CHAR (32)
                                                 /*MEMBER NAME                *
   3 FILLER00013                                 CHAR (70),

 DCL
 1 DPQB_BASED5                                   BASED (DMR_PTR)
   3 FILLER00014                                 CHAR(1),
   3 DPQBTOTL,
     5 FILLER00015                               CHAR(1)
     5 DPQBTOT                                   CHAR(5)   * TOTAL *
     5 FILLER00016                               CHAR(1)
     5 DPQBTMEM                                  CHAR(20)
   3 FILLER00017                                 CHAR (115)

  DCL DMOUT FILE OUTPUT STREAM ENV (F RECSIZE(80)
  DCL ATRACE FILE OUTPUT STREAM ENV (F RECSIZE(80)
 DCL 1 DOUTPUT CHAR(150),
 DCL 1 DCONTRO STATIC EXTERNAL,
      3 DCOMMUNE CHAR (64),
      3 DDMR     CHAR (8) INIT('DM00')
      3 DBUFFLEN FIXED BIN (27) ALIGNED INIT(0),
      3 DINPLEN  FIXED BIN (15) ALIGNED INIT(0),
      3 DINPLREC FIXED BIN (15) ALIGNED INIT(0),
      3 DOUTLEN  FIXED BIN (15) ALIGNED INIT(0),
      3 DOUTLREC FIXED BIN (15) ALIGNED INIT(0),
      3 DFUNC    PIC '9' INIT(1),
      3 DRETURN  PIC '9',
      3 DRETURN1 PIC '9',
      3 DRETURN2 PIC '9',
      3 DSEVRITY CHAR (1),
      3 DOPTION1 PIC '9' INIT(1),
      3 DOPTION2 PIC '9' INIT(1),
      3 DOPTION3 PIC '9',
      3 DOPTION4 PIC '9',
      3 DOPTION5 PIC '9' INIT(1),
```

AD-A188 908 PROTOTYPE REPORT GENERATION FACILITY IN THE DATA TRAFFIC MANAGEMENT SYSTE.. (U) ILLINOIS UNIV AT URBANA DEPT OF COMPUTER SCIENCE I ADIGUZEL ET AL. OCT 87 2/2

UNCLASSIFIED CERL-IR-P-88/02 DACW88-84-C-0011 F/G 5/1 NL




1·0
1·1
1·25
1·4
1·6
1·8
2·0
2·2
2·5
2·8
3·15
3·5
4·0

```
      3 DOPTION6 PIC '9' INIT(1),
      3 DOPTION7 PIC '9' INIT(0),
      3 DOPTION8 PIC '9' INIT(0),
      3 DOPTION9 PIC '9',
      3 DOUTWAIT FIXED DEC (5),
      3 DOUTTOTL FIXED DEC (5),
      3 DOUTTOTM FIXED DEC (5),
      3 DINPNO   FIXED DEC (3),
      3 FILLER00001 CHAR (3),
      3 DINPUT   CHAR (72);


  /*********************/
  /*  MAIN PROCEDURE  */
  /*********************/


  ALLOCATE DPQB_BASED4;
  ALLOCATE DPQB_BASED5;
  HEAD = NULL;
   PUT FILE(ATRACE) EDIT('TMEM= ',TMEM) (COL(1),A,A);
 CALL CATAG(TMEM,HEAD);
  /*
 CALL LISTT(1)
 */
   GO TO ZACSEND;


/*********************************************************************/
/*                                                                   */
/*  PROCEDURE:  CATAG                                                */
/*                                                                   */
/*  PURPOSE:  THIS PROCEDURE CALLS DMRUS FOR EACH KEYWORD            */
/*            IN THE FORMATTED LIST, AND CREATES A LINKED            */
/*            LIST OF EACH MEMBER NAME RETURNED FROM THE             */
/*            DATABASE                                               */
/*                                                                   */
/*********************************************************************/

  CATAG:
     PROCEDURE(STR,PTR);
  DCL STR CHAR(40),
      PTR POINTER,
     T    POINTER;

  PUT FILE(ATRACE) EDIT('INTERNAL KEYWORD IS: ',STR) (COL(1),A,A);
  DINPUT='WHAT FORMS ' CAT STR CAT ' .';
  CALL DMRUS(DCONTRO,DOUTPUT);
  DFUNC=2;
  PTR=NULL;
  CALL DMRUS(DCONTRO,DOUTPUT);
   T = NULL,
   I = 1;
  DO WHILE(DOUTWAIT > 0);
     IF DOUTWAIT NE 1 THEN BEGIN;
      ALLOCATE CATREC;
     CALL DMRUS(DCONTRO,DPQB_BASED4);
     IF (DMTEST = '      ') THEN DO;
     CATPTR->CATREC.CATMEMN=DPQBDMEM;
     CATPTR->CATREC.CATMEMT=DPQBDMTP;
     CATPTR->CATREC.CATNEXT=NULL;
     PUT FILE(ATRACE) EDIT('ITEM = ',CATPTR->CATREC.CATMEMN,'TYPE = ',
          CATPTR->CATREC.CATMEMT)(COL(1),A,A,X(4),A,A);
     IF I > 1 THEN T->CATREC.CATNEXT = CATPTR;
        ELSE PTR = CATPTR;
     T = CATPTR;
     I = I +1;
     END;
     END;
     ELSE BEGIN;
      CALL DMRUS(DCONTRO,DPQB_BASED5);
      TOTALN = DPQBTOT;
     END;
     T=CATPTR;
  END;
   DFUNC = 1;
  END CATAG;


/*********************************************************************/
/*                                                                   */
```

```
/*   PROCEDURE:   LISTT                                                 */
/*                                                                      */
/*   PURPOSE:   THIS PROCEDURE WRITES INTO THE DATA FILE ATRACE         */
/*              THE APPROPRIATE INFORMATION FROM EACH MEMBER            */
/*              OF THE CATEGORY LIST ACCORDING TO THE                   */
/*              SPECIFIC COMMAND USED TO ACCESS THE DATABASE.           */
/*                                                                      */
/************************************************************************/

 LISTT:
     PROCEDURE(TYPE);
       DCL   TYPE FIXED(1);
       DCL   PTR POINTER;

       IF TYPE = 1 THEN BEGIN;
          PTR = HEAD;
          PUT FILE(ATRACE) EDIT('WHAT FORMS--- CLASSIFICATION')(COL(1),A)
          ;
       PUT FILE(ATRACE) EDIT('GOT FOLLOW ITEMS FROM D/D FOR INTERNAL KEY'
                 ) (COL(1),A);
          DO WHILE(PTR NE NULL);
           PUT FILE(ATRACE) EDIT(PTR->CATREC.CATMEMN,PTR->CATREC.CATMEMT
                 ) (COL(1),X(2),A,X(2),A);
             PTR = PTR->CATREC.CATNEXT;
          END;
       END;
       IF TYPE = 2 THEN BEGIN;
          PTR = HEAD;
          PUT FILE(ATRACE) EDIT('WHAT USE---- ACCESS PATH')(COL(1),A);
          DO WHILE (PTR NE NULL);
        PUT FILE(ATRACE) EDIT(PTR->CATREC.CATMEMN,PTR->CATREC.CATMEMT)
        PTR->CATREC.CA
          END;
       END;
       IF TYPE = 3 THEN
          DISPLAY('NOW DISPLAY REPORT DETAILS');
     END LISTT;
ZACSEND:  END ZACSD3;
```

```
YUINIT:
   PROCEDURE OPTIONS(MAIN);

/*
 *  PROJECT:          REPORT GENERATOR
 *  PROGRAMMER:       DEPARTMENT OF COMPUTER SCIENCE
 *                    UNIVERSITY OF ILLINOIS
 *
 *  FILENAME:         INITREPORTDEFNS.PLI
 *  LANGUAGE:         PL/I UNDER CMS
 *  DEPENDENCIES:     SRDEF.DATA
 *
 *  DATE:             DECEMBER, 1984
 */

/*  THIS MODULE CREATES AND INITIALIZES THE INTERNAL REPRESENTATION
    OF THE STANDARD REPORT DEFINITIONS TO BE USED IN THE FRONT END.
*/

        /* STANDARD REPORT DEFINITION FILE */
      DCL SRDEF FILE INPUT SEQUENTIAL ENV(F(1960));
      DCL NULL     BUILTIN;
      DCL DATE      BUILTIN;
      DCL TIME     BUILTIN;
      DCL SUBSTR   BUILTIN;
      DCL INDEX    BUILTIN;

       DCL 1 SREPORT,                    /* SR DEF TEMPLET */
             2 SRID    FIXED(2) INIT(0),    /* REPORT ID */
             2 SFNAME CHAR(32) INIT(''),    /* REPORT NAME */
             2 SDES    CHAR(72) INIT(''),   /* DESCRIPTION */
             2 ST_NUM FIXED(2) INIT(0),     /* NUMBER TARGET ITEMS */
             2 SQ_NUM FIXED(2) INIT(0),     /* NUMBER QUALIFIER ITEMS */
             2 STITEM(10),              /* TARGET & QUALIFIER ITEM LIST */
               3 STINAM CHAR(12) INIT((10)''),   /* ITEM NAME */
               3 SFORMA CHAR(10) INIT((10)''),   /* FORMAT */
               3 SDBNAM CHAR(8) INIT((10)''),    /* DATABASE */
               3 SLOGID CHAR(8) INIT((10)''),    /* LOGON ID */
               3 SPASSW CHAR(8) INIT((10)''),    /* PASSWORD */
               3 SFULNA CHAR(32) INIT((10)''),   /* FULL NAME */
               3 STIDES CHAR(72) INIT((10)''),   /* DESCRIPTION */
               3 SHEIGH FIXED(2) INIT((10)0),    /* LENGTH OF ARRAY PATH */
               3 SPATH(4) CHAR(8) INIT((10)''); /* ARRAY OF PARENT SEGMENTS */
      DCL 1 TITEM,                       /* ITEM RECORD FOR RIF_D */
            2 NAME    CHAR(12) ,
            2 FORMAT  CHAR(10) ,
            2 DBNAME  CHAR(8)  ,
            2 LOGID   CHAR(8)  ,
            2 PASSWD  CHAR(8)  ,
            2 FNAME   CHAR(32) ,
            2 DES     CHAR(72) ,
            2 HEIGHT  PICTURE '99',
            2 PATH(4) CHAR(8);
      DCL 1 RIFREC,                             /* RIF_R RECORD */
            2 SRID     PICTURE '99',            /* REPORT ID */
            2 QUL_NO   PICTURE '99',            /* NUMBER QUALIFIERS */
            2 QUAL(5),
              3 QNAME  CHAR(12),                /* QUALIFIER NAME */
              3 QVALU  CHAR(12),                /* QUALIFIER VALUE */
              3 QVALB  CHAR(12),
              3 QINDEX PICTURE '999',           /* QUALIFIER INDEX */
              3 QFLAG  PICTURE '9',             /* QUALIFIER FLAG */
            2 TI_NO    PICTURE'99',             /* NUMBER TARGET ITEMS */
            2 TINDEX(10) PICTURE'999'; /* INDEX ARRAY TO TARGET ITEMS */

      DCL 1 REPORT(3) STATIC EXTERNAL,      /* WORKING RECORD FOR SR */
            2 FNAME   CHAR(32),             /* FULL NAME */
            2 DES     CHAR(72),             /* DESCRIPTION */
            2 T_NUM   FIXED(2) INIT((3)0),  /* NUMBER TARGET ITEMS */
            2 TLIST   POINTER,              /* TO TARGET ITEM LIST */
            2 Q_NUM   FIXED(2) INIT((3)0),  /* NUMBER OF QUALIFIERS */
            2 QLIST   POINTER,              /* TO QUALIFIER LIST */
            2 FACTOR FIXED(1) INIT((3)0),   /* MATCH FACTOR */
            2 WANTED BIT(1) INIT((3)'0'B); /* WANTED TO BE PRODUCED */
      DCL 1 TARGET BASED(WT),               /* NODE OF TARGET ITEM LIST */
            2 NAME    CHAR(12),
            2 FORMAT  CHAR(10),
            2 DBNAME  CHAR(8),
            2 LOGID   CHAR(8),
```

```
            2 PASSWD CHAR(8),
            2 FNAME  CHAR(32),
            2 DES    CHAR(72),
            2 HEIGHT FIXED(2),
            2 PATH(4) CHAR(8),
            2 MATCH  BIT(1) INIT('0'B),
            2 FWP    POINTER;              /* NEXT TARGET ITEM */
      DCL 1 QUALIF BASED(WQ),              /* NODE OF QUALIFIER LIST */
            2 NAME   CHAR(12),
            2 FORMAT CHAR(10),
            2 DBNAME CHAR(8),
            2 LOGID  CHAR(8),
            2 PASSWD CHAR(8),
            2 FNAME  CHAR(32),
            2 DES    CHAR(72),
            2 HEIGHT FIXED(2),
            2 PATH(4) CHAR(8),
            2 VALU   CHAR(12),
            2 VALB   CHAR(12),
            2 INDEX  FIXED(3),
            2 FLAG   FIXED(1),
            2 MATCH  BIT(1) INIT('0'B),
            2 FWP    POINTER;             /* NEXT QUALIFIER ITEM */

      DCL SP       POINTER;
      DCL S        BIT(1) INIT('1'B);    /* END OF FILE */
      DCL NP       FIXED(2); /* NUMBER TARGET ITEMS ALREADY CREATED */


/*******************/
/  MAIN PROCEDURE */
/*******************/

       I = 1;
       ON ENDFILE (SRDEF) S = '0'B;
       DO WHILE(S);              /* GET SR TEMPLET */
          READ FILE (SRDEF) INTO (SREPORT);
        IF (S) THEN DO;
  /*
          SFNAME = '';
          SDES   = '';
          ST_NUM = 0;
          SQ_NUM = 0;
          DO J=1 TO 10;
           STITEM(J).STINAM = '';
           STITEM(J).SFORMA = '';
           STITEM(J).SDBNAM = '';
           STITEM(J).SLOGID = '';
           STITEM(J).SPASSW = '';
           STITEM(J).SFULNA = '';
           STITEM(J).STIDES = '';
           STITEM(J).SHEIGH = '';
           DO K= 1 TO 4;
            STITEM(J).SPATH(K) = '';
           END;
          END;
  */
          REPORT(I).FNAME = SFNAME;
          REPORT(I).DES = SDES;
          REPORT(I).QLIST = NULL;
          REPORT(I).TLIST = NULL;
          REPORT(I).T_NUM = ST_NUM;
          REPORT(I).Q_NUM = SQ_NUM;
          NP = 1;
          SP = NULL;
          DO J=1 TO ST_NUM;   /* CREATE SR TARGET ITEM LIST */
             ALLOCATE TARGET;
             WT->TARGET.NAME = STITEM(J).STINAM;
            WT->TARGET.FORMAT = STITEM(J).SFORMA;
            WT->TARGET.DBNAME = STITEM(J).SDBNAM;
            WT->TARGET.LOGID  = STITEM(J).SLOGID;
            WT->TARGET.PASSWD = STITEM(J).SPASSW;
            WT->TARGET.FNAME  = STITEM(J).SFULNA;
            WT->TARGET.DES    = STITEM(J).STIDES;
            WT->TARGET.HEIGHT = STITEM(J).SHIEGH;
            DO K=1 TO 4;
             WT->TARGET.PATH(K) = STITEM(J).SPATH(K);
            END;    /* K */
             WT->TARGET.FWP = NULL;
             IF NP > 1 THEN SP->TARGET.FWP = WT;
              ELSE REPORT(I).TLIST = WT;
```

```
                SP = WT;
                NP = NP + 1;
            END;    /* J */
            NP = 1;
            DO J = ST_NUM + 1 TO ST_NUM + SQ_NUM;
                ALLOCATE QUALIF;
                WQ->QUALIF.NAME    = STITEM(J).STINAM;
                WQ->QUALIF.FORMAT  = STITEM(J).SFORMA;
                WQ->QUALIF.DBNAME  = STITEM(J).SDBNAM;
                WQ->QUALIF.LOGID   = STITEM(J).SLOGID;
                WQ->QUALIF.PASSWD  = STITEM(J).SPASSW;
                WQ->QUALIF.FNAME   = STITEM(J).SFULNA;
                WQ->QUALIF.DES     = STITEM(J).STIDES;
                WQ->QUALIF.HEIGHT  = STITEM(J).SHIEGH;
                DO K=1 TO 4;
                 WQ->QUALIF.PATH(K) = STITEM(J).SPATH(K);
                END;   /* K */
                WQ->QUALIF.FWP = NULL;
                IF NP > 1 THEN SP->QUALIF.FWP = WQ;
                 ELSE REPORT(I).QLIST = WQ;
                SP = WQ;
                NP = NP + 1;
            END;    /* J */
          I = I +1;
        END;    /* END IF S */
        END;   /* S */

END YUINIT;
```

```
ZUIF:     PROCEDURE OPTIONS (MAIN);

/*
 *   PROJECT:          REPORT GENERATOR
 *   PROGRAMMER:       DEPARTMENT OF COMPUTER SCIENCE
 *                     UNIVERSITY OF ILLINOIS
 *
 *   FILENAME:         DECISIONMAKER.PLI
 *   LANGUAGE:         PL/I UNDER CMS
 *   DEPENDENCIES:     ACCESSDICTIONARY.PLI
 *
 *   DATE:             DECEMBER, 1984
 */


/* THIS MODULE DETERMINES WHICH STANDARD REPORTS, IF ANY,
   MATCH THE USER'S QUERY, AND THEN PORMPTS THE USER TO
   SELECT EITHER THE STANDARD REPORT(S), OR A NON-STANDARD
   REPORT TO BE PRODUCED.  THE USER ALSO HAS THE CHOICE
   OF NOT GENERATING ANY REPORT.  FOR A STANDARD REPORT,
   THE INFORMATION -- TARGET AND QUALIFIER DATA ITEMS,
   AND REPORT FORMAT(S) -- NEEDED IS PASSED TO THE ACTUAL
   REPORT GENERATOR, I.E., THE BACKEND.          */


      DCL RIFR FILE OUTPUT SEQUENTIAL ENV(F(236)),/* REPORT FORMAT FILE */
          RIFD FILE OUTPUT STREAM ENV(F(184));     /* DATA ITEM FILE */
       DCL ZACSD3 ENTRY EXTERNAL;       /* DICTIONARY ACCESS */
       DCL NULL     BUILTIN;
       DCL DATE      BUILTIN;
       DCL TIME     BUILTIN;
       DCL SUBSTR   BUILTIN;
       DCL INDEX    BUILTIN;

      DCL REPNO DEC FIXED(1) INIT(2); /* RESPONSE NUMBER */
      DCL 1 SREPORT,                  /* SR DEF TEMPLET */
            2 SRID    FIXED(2),       /* REPORT ID */
            2 SFNAME CHAR(32),        /* REPORT NAME */
            2 SDES    CHAR(72),       /* DESCRIPTION */
            2 ST_NUM FIXED(2),        /* NUMBER TARGET ITEMS */
            2 SQ_NUM FIXED(2),        /* NUMBER QUALIFIER ITEMS */
            2 STITEM(10),       /* TARGET & QUALIFIER ITEM LIST */
              3 STINAM CHAR(12),      /* ITEM NAME */
              3 SFORMA CHAR(10),      /* FORMAT */
              3 SDBNAM CHAR(8),       /* DATABASE */
              3 SLOGID CHAR(8),       /* LOGON ID */
              3 SPASSW CHAR(8),       /* PASSWORD */
              3 SFULNA CHAR(32),      /* FULL NAME */
              3 STIDES CHAR(72),      /* DESCRIPTION */
              3 SHEIGH FIXED(2),      /* LENGTH OF ACCESS PATH */
              3 SPATH(4) CHAR(8);     /* ARRAY OF PARENT SEGMENTS */
      DCL 1 TITEM,                    /* ITEM RECORD FOR RIF_D */
            2 NAME    CHAR(12) ,
            2 FORMAT  CHAR(10) ,
            2 DBNAME CHAR(8)   ,
            2 LOGID   CHAR(8)  ,
            2 PASSWD CHAR(8)   ,
            2 FNAME   CHAR(32) ,
            2 DES     CHAR(72) ,
            2 HEIGHT PICTURE '99',
            2 PATH(4) CHAR(8);        /*  ACCESS PATH ARRAY */
      DCL 1 RIFREC,                   /* RIF_R RECORD */
            2 SRID     PICTURE '99',  /* REPORT ID */
            2 QUL_NO   PICTURE '99',  /* NUMBER QUALIFIERS */
            2 QUAL(5),
              3 QNAME   CHAR(12),     /* QUALIFIER NAME */
              3 QVALU   CHAR(12),     /* QUALIFIER VALUE */
              3 QVALB   CHAR(12),
              3 QINDEX PICTURE '999', /* QUALIFIER INDEX */
              3 QFLAG   PICTURE '9',  /* QUALIFIER FLAG */
            2 TI_NO    PICTURE'99',   /* NUMBER TARGET ITEMS */
            2 TINDEX(10) PICTURE'999';
                        /* INDEX ARRAY TO TARGET ITEMS */

      DCL 1 REPORT(3) STATIC EXTERNAL,     /* WORKING RECORD FOR SR */
            2 FNAME   CHAR(32),       /* FULL NAME */
            2 DES     CHAR(72),       /* DESCRIPTION */
            2 T_NUM   FIXED(2),       /* NUMBER OF TARGET ITEMS */
            2 TLIST   POINTER,        /* TO TARGET ITEM LIST */
            2 Q_NUM   FIXED(2),       /* NUMBER QUALIFIER ITEMS */
            2 QLIST   POINTER,        /* TO QUALIFIER LIST */
```

```
        2 FACTOR FIXED(1),          /* MATCH FACTOR */
        2 WANTED BIT(1);            /* WANTED TO BE PRODUCED */
      DCL 1 TARGET BASED(WT),       /* NODE OF TARGET ITEM LIST */
        2 NAME    CHAR(12),
        2 FORMAT  CHAR(10),
        2 DBNAME  CHAR(8),
        2 LOGID   CHAR(8),
        2 PASSWD  CHAR(8),
        2 FNAME   CHAR(32),
        2 DES     CHAR(72),
        2 HEIGHT  FIXED(2),
        2 PATH(4) CHAR(8),
        2 MATCH   BIT(1) INIT('0'B),
        2 FWP     POINTER,            /* NEXT TARGET ITEM */
      DCL 1 QUALIF BASED(WQ),         /* NODE OF QUALIFIER LIST */
        2 NAME    CHAR(12),
        2 FORMAT  CHAR(10),
        2 DBNAME  CHAR(8),
        2 LOGID   CHAR(8),
        2 PASSWD  CHAR(8),
        2 FNAME   CHAR(32),
        2 DES     CHAR(72),
        2 HEIGHT  FIXED(2),
        2 PATH(4) CHAR(8),
        2 VALU    CHAR(12),
        2 VALB    CHAR(12),
        2 INDEX   FIXED(3),
        2 FLAG    FIXED(1),
        2 MATCH   BIT(1) INIT('0'B),
        2 FWP     POINTER;                  /* NEXT NODE */

      DCL (S,RID)        BIT(1) INIT('1'B);  /* REPORT ID */

      DCL RK POINTER;                        /* TO QUERY LIST */
      DCL (TGHEAD,QUHEAD) POINTER STATIC EXTERNAL;
                    /* TARGET AND QUALIFIER HEAD OF LIST */
        DCL FORMLENG DEC FIXED(2) INIT(40),/* FORMAT LENGTH */
            FORMWD(50) CHAR(FORMLENG) VARYING;
                    /* FORMAT FOR DATA DICTIONARY ACCESS */
      DCL (SN1,SN2,SN3) BIT(1) INIT('0'B);
                    /* EXACT MATCH, PARTIAL MATCH, NO MATCH */
      DCL CASE_NO  FIXED(1),
          PN(5)    FIXED(1);
              /* PN(1) - NUMBER REPORTS WITH MATCH FACTOR - 1 */
      DCL WANT      CHAR(10) VARYING INIT(''),
      DCL YESORNO   CHAR(4) INIT('');

      DCL 1 YTARGET BASED(CURR_TARGET_PT),/* NODE OF TARGET ITEM LIST */
        2 STCOUNT   FIXED(2),             /* NUMBER OF ASSOCIATED */
                                          /* INTERNAL KEYWORDS */
        2 MATCH BIT(1),
        2 STGPTR    POINTER,              /* TO INTERNAL KEYWORD LIST */
        2 NEXTTG    POINTER,              /* NEXT TARGET ITEM */
      DCL 1 YQUALI BASED(CURR_QUALI_PT),/* NODE OF QUALIFIER ITEM LIST */
        2 SQCOUNT   FIXED(2),             /* NUMBER OF ASSOCIATED */
                                          /* INTERNAL KEYWORDS */
        2 QVALUEU CHAR(12),               /* QUALIFIER VALUE */
        2 QVALUEB CHAR(12),
        2 MATCH BIT(1),
        2 SQUPTR    POINTER,              /* TO INTERNAL KEYWORD LIST */
        2 NEXTQU    POINTER,              /* NEXT QUALIFIER ITEM */
      DCL 1 YSKWDL BASED(CURR_SKWD_PT),   /* USER SEARCH KEYWORD RECORD */
        2 IKCOUNT DEC FIXED(2),           /* NUMBER INTERNAL KEYWORDS */
        2 IKPTR POINTER,                  /* NEXT INTERNAL KEYWORD */
        2 NEXTSK POINTER,                 /* NEXT USER KEYWORD */
      DCL 1 INKWDREC  BASED(CURR_INKWDREC_PT),/* INTERNAL KEYWORD RECORD */
        2 INKWDF      CHAR(12) VARYING,   /* INTERNAL KEYWORD */
        2 NEXTINKWD     POINTER,          /* NEXT INTERNAL KEYWORD */
      DCL I INDEX DEC FIXED(5) INIT(1),
    /* COUNT OF THE TOTAL NUMBER OF TARGET AND QUALIFIER ITEMS PROCESSED */

      DCL I FIXED(1),


/*********************/
/*  MAIN PROCEDURE  */
/*********************/


DO    IF TGHEAD = NULL THEN BEGIN,        /* NO TARGET ITEM GIVEN */
        PUT SKIP,
```

```
          DISPLAY('IF YOU NEED HELP, TYPE HELP');
          DISPLAY('OTHERWISE, TYPE RETURN KEY TO ENTER QUERY AGAIN')
             REPLY(WANT);
          IF WANT='HELP' THEN CALL HELP;
           ELSE IF WANT='' THEN GO TO UIFEND;
           ELSE BEGIN;
                 DISPLAY('BAD INPUT, DO IT AGAIN');
                 GO TO D0;
                END;
          GO TO DEND;
         END;
D1:    DO I=1 TO REPNO;                    /* COUNT MATCH FACTOR */
        CALL MTSR(I);                      /* COUNT MATCH FACTOR */
        IF SN1 THEN REPORT(I).FACTOR=1;          /* EXACT MATCH */
        ELSE IF SN2 THEN REPORT(I).FACTOR=2; /* PARTIAL MATCH */
        ELSE REPORT(I).FACTOR=3;                 /* NO SR MATCHED */
       END D1;
D2:    PN=0;                    /* COUNT NUMBER OF MATCHED REPORTS */
       DO I=1 TO REPNO;
          IF REPORT(I).FACTOR=1 THEN PN(1)=PN(1)+1;
          IF REPORT(I).FACTOR=2 THEN PN(2)=PN(2)+1;
          IF REPORT(I).FACTOR=3 THEN PN(3)=PN(3)+1;
       END;
       IF PN(1) > 0 THEN BEGIN;
                      CALL CHOS(1,PN(1));
                      GO TO DEND;
                     END;
       ELSE IF PN(2) > 0 THEN BEGIN;
                         CALL CHOS(2,PN(2));
                         GO TO DEND;
                        END;
       ELSE IF PN(3) > 0 THEN BEGIN;
                         CALL CHOS(3,PN(3));
                         GO TO DEND;
                        END;
       ELSE CALL ERROR_HANDLE1;
DEND:  ;


/*****************************************************************/
/*                                                               */
/*   PROCEDURE: CHOS                                             */
/*                                                               */
/*   PURPOSE:   THIS PROCEDURE INFORMS THE USER WHETHER          */
/*              AND TO WHAT DEGREE HIS QUERY MATCHES ANY         */
/*              ANY OF THE STANDARD REPORTS.  IT THEN            */
/*              PROMPTS THE USER TO SELECT EITHER ONE OR         */
/*              MORE REPORTS OF THE DISPLAYED REPORTS,           */
/*              OR A NON-STANDARD REPORT                         */
/*                                                               */
/*****************************************************************/


CHOS   PROCEDURE(FACTOR,PN);
       DCL (FACTOR,PN,I,J) FIXED(1);
       DCL RID             BIT(1);
       DCL SRF             BIT(1);
       DCL CASE(3)         LABEL;

        RID = '1'B,            /* SR */
        GO TO CASE(FACTOR);
       /* CHOOSE EXACTLY MATCHED SR */
CASE(1): I=1;
 PUT SKIP EDIT ('FOLLOWING STANDARD REPORT(S) SATISFY YOUR QUERY') (A);
          DO J=1 TO REPNO;
             IF REPORT(J).FACTOR=1 THEN BEGIN;
                CALL LTSR(J,I);
                I = I + 1;
                IF I > PN THEN GO TO C1;
               END;
            END;
C1:        CALL SURE(1,SRF);
           IF SRF='1'B THEN CALL WRIF(RID);
           GO TO CEND;
       /* CHOOSE PARTIALY MATCHED SR */
CASE(2): I=1;
           PUT SKIP;
         PUT SKIP EDIT('THERE ARE NO STANDARD REPORTS ')(A);
         PUT SKIP EDIT ('WHICH EXACTLY MEET YOUR REQUEST') (A);
         PUT SKIP EDIT ('THE FOLLOWING STANDARD REPORTS') (A);
         PUT SKIP EDIT ('ALL PARTIALLY SATISFY YOUR QUERY') (A);
```

```
            DO J=1 TO REPNO;
               IF REPORT(J).FACTOR=2 THEN BEGIN;
                  CALL LTSR(J,I);
                  I = I + 1;
                  IF I > PN THEN GO TO C2;
                END;
            END;
C2:         CALL SURE(2,SRF);
            IF SRF='1'B THEN
               CALL WRIF(RID);
            ELSE BEGIN;
               PUT EDIT ('DO YOU WANT TO GENERATE A NON-STANDARD')
                     (A);
               PUT SKIP EDIT ('REPORT?  (YES/NO)') (A);
               CALL YORN(YESORNO);
               IF YESORNO='YES' THEN DO;
                  RID = '0'B;
                  CALL WRIF(RID);
                 END;
                ELSE GO TO UIFEND;
             END;
            GO TO CEND;
 CASE(3):  DISPLAY('NO STANDARD REPORTS SATISFY YOUR QUERY');
            DISPLAY ('WOULD YOU LIKE TO HAVE A NON-STANDARD REPORT?');
            DISPLAY ('YES/NO');
            CALL YORN(YESORNO);
            IF YESORNO='YES' THEN DO;
               RID = '0'B;
               CALL WRIF(RID);
               END;
               ELSE GO TO UIFEND;
CEND:    END CHOS;


/*************************************************************/
/*                                                           */
/*   PROCEDURE: MTSR                                         */
/*                                                           */
/*   PURPOSE:   THIS PROCEDURE DETERMINES WHICH STANDARD     */
/*              REPORTS MATCH THE USER'S REQUEST BY          */
/*              COMPARING EACH TARGET AND QUERY ITEM OF      */
/*              EACH STANDARD REPORT TO THE INTERNAL         */
/*              KEYWORD LIST ASSOCIATED WITH THE USER'S      */
/*              REQUEST   A MATCH FACTOR IS THEN             */
/*              ASSIGNED TO EACH STANDARD REPORT BASED       */
/*              ON THE FOLLWOING RULES:                      */
/*                FACTOR = 1, AN EXACT MATCH                 */
/*                FACTOR = 2, A PARTIAL MATCH                */
/*                FACTOR = 3, IF NO MATCH AT ALL             */
/*                                                           */
/*************************************************************/

MTSR:  PROCEDURE(P);
       DCL P    FIXED(1);
       DCL (TC,QC) FIXED(2),/* NUMBER MATCHED TARGET & QUALIFER ITEMS */
       DCL K    FIXED(1);          /* NUMBER QUERY ITEMS */
       DCL (RL,SL,KL) POINTER INIT(NULL);  /* TO TRAVERSE LISTS */

M0:    TC=0;
       QC = 0;
       SN1='0'B;
       SN2 = '0'B;
       SN3 = '0'B;
       K = 0;
       RL = TGHEAD;
       DO WHILE( RL NE NULL );     /* COUNT NO. OF QUERY ITEMS */
          K = K + 1;
          RL = RL->NEXTTG;
       END;
M1:    SL = REPORT(P).TLIST;     /* TRY TO MATCH TARGET ITMES */
       DO WHILE ( SL NE NULL ), /* FOR EACH TARGET ITEM OF SR */
        RL = TGHEAD;
        SL->TARGET.MATCH = '0'B;
        DO WHILE ( RL NE NULL ); /* FOR TARGET ITEM IN QUERY LIST */
         KL = RL->STGPTR;
         DO WHILE (KL NE NULL); /* FOR EACH INTERNAL KEYWORD */
            IF SL->TARGET.NAME = KL->INKWDF THEN
             BEGIN;
              SL->TARGET.MATCH = '1'B;
              TC = TC + 1;
              GO TO M1A;
```

```
            END;
           KL = KL->NEXTINKWD;
          END;
          RL = RL->NEXTTQ;
        END;
M1A:    SL = SL->TARGET.FWP;
       END;
M2:    IF (TC=REPORT(P).T_NUM) & (K=REPORT(P).T_NUM) THEN SN1='1'B;
       ELSE IF (TC > 0) & ( K < REPORT(P).T_NUM) THEN SN1='1'B;
       ELSE IF TC > 0 THEN SN2='1'B;
       ELSE SN3='1'B;
                    /* TRY TO MATCH QUALIFIER LIST */
                    /* IF MATCH, SET FLAG AND COPY THE VALUES */
M3:    SL = REPORT(P).QLIST;
       DO WHILE (SL NE NULL);      /* FOR EACH QUALIFIER OF SR */
       RL = QUHEAD;
       SL->QUALIF.MATCH = '0'B;
       SL->QUALIF.FLAG = 0;
       SL->QUALIF.VALU = '';
       SL->QUALIF.VALB = '';
        DO WHILE (RL NE NULL);     /* FOR QUALIFIER IN QUERY LIST */
         KL = RL->SQUPTR;
         DO WHILE (KL NE NULL); /* FOR EACH INTERNAL KEYWORD */
            IF SL->QUALIF.NAME = KL->INKWDF THEN
              BEGIN;
              SL->QUALIF.MATCH = '1'B;
              SL->QUALIF.FLAG = 1;
              SL->QUALIF.VALU = RL->QVALUEU;
              SL->QUALIF.VALB = RL->QVALUEB;
               QC = QC + 1;
               GO TO M3A;
             END;
            KL = KL->NEXTINKWD;
           END;
          RL = RL->NEXTQU;
        END;
M3A:    SL = SL->QUALIF.FWP;
      END;
     END MTSR;


/*************************************************************/
/*                                                           */
/*   PROCEDURE: SURE                                         */
/*                                                           */
/*   PURPOSE:   THIS PROCEDURE REQUIRES THE USER TO          */
/*              VERIFY OR CONFIRM HIS EARLIER SELECTION      */
/*              OF THE SPECIFIC REPORT(S) TO BE              */
/*              GENERATED.                                   */
/*                                                           */
/*************************************************************/


SURE:  PROCEDURE(F,SRF);
       DCL F       FIXED(1),
           SRF     BIT(1);
       DCL A       CHAR(1),
           C       CHAR(REPNO*2),
           SRID    CHAR(20),
           I       FIXED(2),
           N       FIXED(1);
       DCL FLAG BIT(1),
           IND BIT(1),
           BAD_REQ BIT(1),
           YES BIT(1) INIT('1'),
           NO BIT(1) INIT('0');

     C='12';
     DO I=1 TO REPNO;
        IF REPORT(I).FACTOR NE F THEN
           SUBSTR(C,I,1) = '0';
      END;
     FLAG=YES;
     IND=YES;
     BAD_REQ = NO;
   DO WHILE (FLAG);
  S0:
      IF BAD_REQ = NO THEN
        GOTO S1;

      BAD_REQ = NO;
```

```
      PUT SKIP EDIT ('THE FOLLOWING REPORT(S) SATISFY YOUR QUERY:')
         (A);
       DO I=1 TO REPNO,
          IF (REPORT(I).FACTOR = F) THEN
              PUT EDIT(I) (F(3));
       END,

S1:    DO I=1 TO REPNO;
          REPORT(I).WANTED = '0'B,
       END;
S1A:   PUT SKIP;
       PUT SKIP;
       DISPLAY('WHICH OF THE ABOVE REPORT(S) DO YOU WANT?');
       SRID='';
       DISPLAY('          TYPE WANTED REPORT ID (E.G. 1,2,...)');
       IF (IND) THEN DO;
          DISPLAY ('          OR TYPE ? FOR MORE DESCRIPTION ');
          DISPLAY ('          OF THE REPORT(S)');
          IND=NO;
       END,
       DISPLAY('          OR, TYPE RETURN KEY IF YOU DON''T LIKE ANY ');
       DISPLAY('          OF THESE REPORTS')
       REPLY(SRID),
       PUT SKIP,
       SELECT(SRID),
            WHEN ('QUIT') DO;
                PUT SKIP;
              PUT SKIP EDIT('REPORT GENERATOR EXITED AT USER''S REQUEST')
                  (A);
                STOP,
            END,
            WHEN('') DO,
             SRF='0'B,
             FLAG=NO;
           END;
          WHEN('?') DO;
          J=1,
        DO I=1 TO REPNO;
            IF REPORT(I).FACTOR = F THEN DO;
               CALL LTSRF(I,J);
               J=J+1;
            END;
        END;
      END;
          OTHERWISE DO;
       DO I=1 TO 20;              /* CHECK REQUEST */
         A = ' ',
         A = SUBSTR(SRID,I,1);
         IF ((A NE ' ') AND (A NE ',')) THEN
            BEGIN;
              N = INDEX(C,A);
              IF N=0 THEN
               BEGIN;
                 PUT SKIP EDIT('REPORT ID  ', A, '  IS NOT ') (A,A,A);
                 PUT SKIP EDIT('ONE OF YOUR ABOVE SPECIFIED OPTIONS.')
                    (A),
                 BAD_REQ = YES;
               END;
              ELSE REPORT(N).WANTED = '1'B;
            END;
           END;
       IF BAD_REQ THEN
         BEGIN;
           PUT SKIP;
           PUT SKIP EDIT ('TRY IT AGAIN') (A);
           PUT SKIP,
           GOTO S0;
        END;
      PUT SKIP;
S2    DISPLAY('YOU HAVE SELECTED THE FOLLOWING STANDARD REPORTS');
      DISPLAY ('FOR GENERATION');
       J=1,
       DO I=1 TO REPNO,
          IF REPORT(I) WANTED='1'B THEN DO;
             CALL LTSR(I,J);
             J = J + 1,
           END,
       END;
       PUT SKIP,
       DISPLAY('IS THIS LISTING CORRECT ?');
       CALL YORN(YESORNO);
```

```
      IF YESORNO='YES' THEN
        BEGIN;
          SRF = '1'B;
          FLAG=NO;
        END;
    END; /*END OTHERWISE */
    END; /* END SELECT */
  END; /* WHILE FLAG */
SEND: END SURE;

/*****************************************************************/
/*                                                               */
/*   PROCEDURE: LTSR                                             */
/*                                                               */
/*   PURPOSE:   THIS PROCEDURE DISPLAYS ON THE USER'S            */
/*              TERMINAL A LIST OF THE MATCHING                  */
/*              STANDARD REPORTS.                                */
/*                                                               */
/*****************************************************************/

LTSR: PROCEDURE(RID,P);
      DCL (RID,P) FIXED(1);

      IF P=1 THEN DO;
         PUT SKIP EDIT ('REPORT ID','REPORT FULL NAME') (R(FM1));
         PUT EDIT((45)'-') (COL(1),A(45));
      END;
      PUT SKIP EDIT (RID,REPORT(RID).FNAME) (R(FM2));
FM1:  FORMAT(X(2),A(9),X(2),A);
FM2:  FORMAT(X(2),F(9),X(2),A);
      END LTSR;


/*****************************************************************/
/*                                                               */
/*   PROCEDURE: LTSRF                                            */
/*                                                               */
/*   PURPOSE:   THIS PROCEDURE DISPLAYS ON THE USER'S            */
/*              TERMINAL A LIST OF THE MATCHING                  */
/*              STANDARD REPORTS AND THEIR DESCRIPTIONS.         */
/*                                                               */
/*****************************************************************/

 LTSRF:
    PROCEDURE(RID,P);
 DCL (RID,P) FIXED(1);
 DCL DESTMP CHAR(72),
     DESC(72) CHAR(1) DEFINED DESTMP;

 DESTMP=REPORT(RID).DES;
 IF (P=1) THEN DO;
      PUT SKIP EDIT('REPORT ID','REPORT FULL NAME',
          'REPORT DESCRIPTION') (COL(1),X(2),A(9),X(10),A(16),X(10),A);
       PUT EDIT((72)'-') (COL(1),A(72));
    END;
    PUT EDIT(RID,REPORT(RID).FNAME,DESC(1))
        (COL(1),X(6),F(2),X(5),A(32),X(2),A(1));
    DO J=2 TO 25;
       PUT EDIT(DESC(J))(A(1));
    END;
    DO I= 1 TO 2;
    IF (SUBSTR(DESTMP,26+(I-1)*25,25) NE (25)' ') THEN DO;
         PUT SKIP;
         PUT EDIT(' ') (COL(47),A(1));
       K=26+I*25-1;
        IF (K GT 72) THEN K=72;
        DO J=26+(I-1)*25 TO K;
           PUT EDIT(DESC(J)) (A(1));
        END;
      END;
      END;
 END LTSRF;


/*****************************************************************/
/*                                                               */
/*   PROCEDURE: YORN                                             */
/*                                                               */
/*   PURPOSE:   THIS PROCEDURE PROMPTS THE USER TO INPUT         */
/*              'YES', 'NO' OR 'QUIT'.                           */
```

```
/*                                                              */
/*..............................................................*/

YORN: PROCEDURE(YN);
      DCL YN CHAR(4);
      YN = '';
      DO WHILE(((YN NE 'YES') AND (YN NE 'NO')) AND (YN NE 'QUIT'));
         DISPLAY('TYPE YES/NO') REPLY(YN);
       END;

      IF YN = 'QUIT' THEN
        DO;
          PUT SKIP;
          PUT SKIP;
           PUT SKIP EDIT ('REPORT GENERATOR EXITED AT USER''S REQUEST')
                 (A);
          STOP;
        END;
      END YORN;


/*..............................................................*/
/*                                                              */
/*   PROCEDURE:  ERROR_HANDLE1                                  */
/*                                                              */
/*   PURPOSE:    THIS PROCEDURE WILL EVENTUALLY DEAL            */
/*               WITH ERRORS.                                   */
/*                                                              */
/*..............................................................*/

ERROR_HANDLE1:
       PROCEDURE;
       DISPLAY('NOW DEAL WITH ERRORS');
       END;


/*..............................................................*/
/*                                                              */
/*   PROCEDURE:  WRIF                                           */
/*                                                              */
/*   PURPOSE:    THIS PROCEDURE WRITES THE TARGET AND           */
/*               QUALIFIER DATA ITEMS ASSOCIATED WITH THE       */
/*               REQUESTED REPORT(S) INTO THE FILE RIFD         */
/*               IT ALSO WRITES THE REPORT(S) FORMAT INTO       */
/*               THE FILE RIFR.  IN THE CASE OF                 */
/*               NON-STANDARD REPORTS ...                       */
/*                         NOT YET IMPLEMENTED.                 */
/*                                                              */
/*..............................................................*/

WRIF: PROCEDURE(RID);
      DCL RID      BIT(1),
          P        POINTER,
          FILENAM  CHAR(6),
          (I,J,K)   FIXED(2);
      DCL ONCODE   BUILTIN;

 OPEN FILE(RIFR),FILE(RIFD);
      ON RECORD (RIFR)
       BEGIN;
         PUT SKIP;
         PUT SKIP EDIT('***** RECORD CONDITION RAISED IN WRITING
             FILE ***** ',FILENAM)(A,A(6));
         PUT SKIP EDIT('       REPORT ',I,' IGNORED')
             (A,F(1),A);
        PUT SKIP EDIT('       ON CODE = ',ONCODE)(A,F(2));
       END;
W1:   IF RID='1'B THEN          /* SR */
       BEGIN;
        DO I=1 TO REPNO;
         IF REPORT(I).WANTED='1'B THEN
          BEGIN;
           /* MOVE REPORT DATA TO TITEM */
           P = REPORT(I).TLIST;
           J = 0;
           DO K=1 TO 10;
              TINDEX(K) = '';
           END;
           DO WHILE(P NE NULL);
              /* MOVE TARGET ITEM DATA TO TITEM RECORD */
```

```
                TITEM.NAME    = P->TARGET.NAME;
                TITEM.FORMAT  = P->TARGET.FORMAT;
                TITEM.DBNAME  = P->TARGET.DBNAME;
                TITEM.LOGID   = P->TARGET.LOGID;
                TITEM.PASSWD  = P->TARGET.PASSWD;
                TITEM.FNAME   = P->TARGET.FNAME;
                TITEM.DES     = P->TARGET.DES;
                TITEM.HEIGHT  = P->TARGET.HEIGHT;
                TITEM.PATH    = P->TARGET.PATH;
        PUT FILE(RIFD) EDIT(TITEM.NAME) (COL(1),A(12));
        PUT FILE(RIFD) EDIT(TITEM.FORMAT) (A(10));
        PUT FILE(RIFD) EDIT(TITEM.DBNAME) (A(8));
        PUT FILE(RIFD) EDIT(TITEM.LOGID) (A(8));
        PUT FILE(RIFD) EDIT(TITEM.PASSWD) (A(8));
        PUT FILE(RIFD) EDIT(TITEM.FNAME) (A(32));
        PUT FILE(RIFD) EDIT(TITEM.DES) (A(72));
        PUT FILE(RIFD) EDIT(TITEM.HEIGHT) (F(2));
        PUT FILE(RIFD) EDIT(TITEM.PATH) (4 A(8));
                 FILENAM = 'RIFD';
                TINDEX(J + 1 ) = IINDEX;
                IINDEX = IINDEX + 1;
                J = J + 1;
                P = P->TARGET.FWP;
            END;
            P = REPORT(I).QLIST;
            J = 1;
            DO K=1 TO 5;
                QUAL(K).QNAME = '';
                QUAL(K).QVALU = '';
                QUAL(K).QVALB = '';
                QUAL(K).QFLAG = 0;
                QUAL(K).QINDEX ='';
            END;
            DO WHILE(P NE NULL);
                /* MOVE QUALIFIER DATA TO TITEM RECORD */
                TITEM.NAME    = P->QUALIF.NAME;
                TITEM.FORMAT  = P->QUALIF.FORMAT;
                TITEM.DBNAME  = P->QUALIF.DBNAME;
                TITEM.LOGID   = P->QUALIF.LOGID;
                TITEM.PASSWD  = P->QUALIF.PASSWD;
                TITEM.FNAME   = P->QUALIF.FNAME;
                TITEM.DES     = P->QUALIF.DES;
                TITEM.HEIGHT  = P->QUALIF.HEIGHT;
                TITEM PATH    = P->QUALIF.PATH;

        PUT FILE(RIFD) EDIT(TITEM.NAME) (COL(1),A(12));
        PUT FILE(RIFD) EDIT(TITEM.FORMAT) (A(10));
        PUT FILE(RIFD) EDIT(TITEM.DBNAME) (A(8));
        PUT FILE(RIFD) EDIT(TITEM.LOGID) (A(8));
        PUT FILE(RIFD) EDIT(TITEM.PASSWD) (A(8));
        PUT FILE(RIFD) EDIT(TITEM.FNAME) (A(32));
        PUT FILE(RIFD) EDIT(TITEM.DES) (A(72));
        PUT FILE(RIFD) EDIT(TITEM.HEIGHT) (F(2));
        PUT FILE(RIFD) EDIT(TITEM.PATH) (4 A(8));
                /* MOVE QUAL DATA TO RIFREC */
                QUAL(J).QNAME = P->QUALIF.NAME;
                QUAL(J).QVALU = P->QUALIF.VALU;
                QUAL(J).QVALB = P->QUALIF.VALB;
                QUAL(J).QFLAG = P->QUALIF.FLAG;
                QUAL(J).QINDEX = IINDEX;
                IINDEX = IINDEX + 1;
                J = J + 1;
                P = P->QUALIF.FWP;
            END;
            /* MOVE REPORT DATA TO RIFREC RECORD */
            IF REPORT(I).T_NUM = 0 THEN GO TO W3;
            RIFREC.SRID = 1;
            RIFREC.TI_NO = REPORT(I).T_NUM;
            RIFREC.QUL_NO = REPORT(I).Q_NUM;
            FILENAM = 'RIFR';
            WRITE FILE (RIFR) FROM (RIFREC);
 PUT SKIP;
          END;    /* REPORT(I) */
        END;
      END;       /* SR */
W3:   IF RID='0'B THEN DO;
        DISPLAY('NOW BRANCH TO, AS OF YET, UNIMPLEMENTED ');
        DISPLAY ('NON-STANDARD REPORT GENERATOR');
      END;
        /* MOVE QUERY LIST DATA TO RIFREC AND TITEM RECORD */
        /* THIS PORTION OF THE PROCEDURE IS TO BE DEVELOPED */
```

```
W3:    PUT SKIP;
 CLOSE FILE(RIFR),FILE(RIFD);
  END WRIF;


/*****************************************************************/
/*                                                               */
/*  PROCEDURE:  HELP                                             */
/*                                                               */
/*  PURPOSE:  THIS PROCEDURE LISTS ALL THE STANDARD              */
/*            REPORTS CURRENTLY AVAILABLE IN THE                 */
/*            SYSTEM, AND DISPLAYS SOME EXAMPLE QUERIES          */
/*                                                               */
/*****************************************************************/

HELP:  PROCEDURE;
       DCL I    FIXED(2);
       PUT SKIP;
      PUT SKIP EDIT('THE FOLLOWING STANDARD REPORTS ARE AVAILABLE:')
         (A);
       DO I=1 TO REPNO;
        CALL LTSR(I,1);
       END;
       PUT SKIP,
       PUT SKIP EDIT('QUERY EXAMPLES INCLUDE:') (A);
       PUT SKIP;
       PUT SKIP EDIT('     TELL ME SOMETHING ABOUT COST')(A);
       PUT SKIP EDIT('     SHOW THE COST FOR PROJECT=B12 AND YEAR=1981')
                  (A);
       PUT SKIP EDIT('     REPORT COST WITH PROJECT=B12 AND YEAR=1981')
                  (A);
   END HELP;

UIFEND: END ZUIF;
```

```
QUAL:     PROCEDURE OPTIONS (MAIN);

/*
 *   PROJECT:          REPORT GENERATOR
 *   PROGRAMMER:       DEPARTMENT OF COMPUTER SCIENCE
 *                     UNIVERSITY OF ILLINOIS
 *
 *   FILENAME:         TESTQUALIFIED.PLI
 *   LANGUAGE:         PL/I UNDER CMS
 *   DEPENDENCIES:     RIFR.DATA
 *                     RIFD.DATA
 *
 *   DATE:             DECEMBER, 1984
 */

/*   THIS MODULE READS IN AND EXAMINES THE REPORT FORMAT(S) AND
     THE ASSOCIATED QUALIFIER AND TARGET ITEMS CREATED BY THE
     FRONT END.  IT PROMPTS THE USER TO FILL IN ANY UNSPECIFIED
     QUALIFIERS, AND THEN SENDS THE LIST OF REQUESTED REPORTS
     TO THE REPORT GENERATOR VIA THE FILE INDATA.DATA
*/


   DCL RIFR FILE UPDATE SEQUENTIAL ENV(F(236)),
       RIFD FILE INPUT SEQUENTIAL ENV(F(184)),
       INDATA FILE OUTPUT SEQUENTIAL ENV(F(80));

   DCL 1 STITEM(15),                  /* TARGET & QUALIFIER ITEMS */
         2 STINAM CHAR(12),           /* ITEM NAME */
         2 SFORMA,                    /* FORMAT */
           3 SFORMA1 CHAR,
           3 SFORMA2 CHAR(9),
         2 SDBNAM CHAR(8),            /* DATABASE */
         2 SLOGID CHAR(8),            /* LOGON ID */
         2 SPASSW CHAR(8),            /* PASSWORD */
         2 SFULNA CHAR(32),           /* FULL NAME */
         2 STIDES CHAR(72),           /* DESCRIPTION */
         2 SHEIGH FIXED(2),           /* LENGTH OF ACCESS PATH */
         2 SPATH(4) CHAR(8);          /* ARRAY OF PARENT SEGMENTS */
   DCL 1 RIFREC,                      /* RIF_R RECORD */
        2 SRID     PICTURE '99',      /* REPORT ID */
        2 QUL_NO   PICTURE '99',      /* NUMBER OF QUALIFIERS */
        2 QUAL(5),
          3 QNAME  CHAR(12),          /* QUALIFIER NAME */
          3 QVALU(12) CHAR,           /* QUALIFIER VALUE */
          3 QVALB  CHAR(12),
          3 QINDEX PICTURE '999',     /* QUALIFIER INDEX */
          3 QFLAG  CHAR,              /* QUALIFIER FLAG */
        2 TI_NO    PICTURE'99',      /* NUMBER OF TARGET ITEMS */
        2 TINDEX(10) PICTURE'999';   /* INDEX ARRAY TO TARGET ITEMS */

   DCL 1 INREC,                        /* INTERNAL DATA RECORD */
         2 SRID PICTURE '99',          /* REPORT ID */
         2 DUMMY CHAR(78) INIT (' ');

   DCL INVALU CHAR(12) INIT (' ');
   DCL INVALU1(12) CHAR INIT (' ');

   DCL FORMA2 CHAR(9) VARYING;
   DCL J FIXED(2);
   DCL LEN4 FIXED(2);
   DCL I FIXED(2);
   DCL K FIXED(2);
   DCL CNT FIXED DECIMAL(2);          /* NUMBER OF MISSING QUALIFIERS */
   DCL UNFILL CHAR(5) VARYING;        /* FLAG -- SIGNALS FILLED/UNFILLED */
   DCL EMPTY CHAR(5) VARYING;         /* FLAG -- SIGNALS EMPTY/NONEMPTY */
   DCL CCNT FIXED(2);
   DCL REM(10) FIXED(2);     /* ARRAY OF INDICES TO MISSING QUALIFIERS */
   DCL REMQ(10) FIXED(2);    /* QUALIFIER NUMBER ARRAY FOR REM */


/*********************/
/*  MAIN PROCEDURE   */
/*********************/

   OPEN FILE(RIFR), FILE(INDATA);
   READ FILE(RIFR) INTO (RIFREC);
   INREC.SRID = RIFREC.SRID;
   WRITE FILE(INDATA) FROM (INREC);
   IF (INREC.SRID <= 0)
     THEN RETURN;
```

```
   OPEN FILE(RIFD);
   DO I = 1 TO TI_NO + QUL_NO;
      READ FILE(RIFD) INTO (STITEM(I));
   END;
   UNFILL = 'FALSE';
   CNT = 0;
   DO I = 1 TO QUL_NO;
      IF (QFLAG(I) = '0')
        THEN DO;
          CNT = CNT + 1;
          REM(CNT) = RIFREC.QUAL(I).QINDEX;
          REMQ(CNT) = I;
          UNFILL = 'TRUE';
        END;
      END;
   IF UNFILL = 'TRUE'
      THEN CALL FILL;
   RETURN;

/*****************************************************************/
/*                                                               */
/*  PROCEDURE:  FILL                                             */
/*                                                               */
/*  PURPOSE:  THIS PROCEDURE PROMPTS THE USER FOR THE            */
/*            QUALIFIERS MISSING FROM HIS REQUESTED REPORT(S).   */
/*                                                               */
/*****************************************************************/

   FILL:  PROCEDURE;

     PUT SKIP EDIT ('TO COMPLETE THE QUERY, YOU MUST SPECIFY THE ',
       'VALUES OF THE FOLLOWING DATA ITEMS') (A,A);
     DO I = 1 TO CNT;
        J = REM(CNT);
        FORMA2 = STITEM(J).SFORMA.SFORMA2;
        LEN4 = LENGTH(FORMA2);
        PUT SKIP EDIT(STITEM(J).STINAM, ' IN ', STITEM(J).SDBNAM,
                      ' FORMAT = ', STITEM(J).SFORMA.SFORMA1, FORMA2)
                          (A(12), A, A(8), A, A, A(LEN4));
        PUT SKIP EDIT ('DESC:', STITEM(J).STIDES) (A, A(72));
        GET EDIT(INVALU) (A(12));
        DO K = 1 TO 12;
          INVALU1(K) = SUBSTR(INVALU, K, 1);
        END;
        IF (STITEM(J).SFORMA.SFORMA1 = 'I')
          THEN DO;
            CCNT = 12;
            EMPTY = 'FALSE';
            DO K = 1 TO 12 WHILE (EMPTY = 'FALSE') ;
              IF INVALU1(K) = ' '
                THEN DO;
                  CCNT = K-1;
                  EMPTY = 'TRUE';
                END;
            END;
            IF (CCNT < 12)
              THEN DO;
                DO K = 0 TO CCNT-1;
                  INVALU1(12-K) = INVALU1(CCNT -K);
                END;
                DO K = 1 TO 12-CCNT;
                  INVALU1(K) = ' ';
                END;
              END;
            DO K = 1 TO 12;
              RIFREC.QUAL(REMQ(I)).QVALU(K) = INVALU1(K);
            END;
          END;
          ELSE DO;
            DO K = 1 TO 12;
              RIFREC.QUAL(REMQ(I)).QVALU(K) = INVALU1(K);
            END;
          END;
        RIFREC.QUAL(REMQ(I)).QFLAG = '1';
     END;
     REWRITE FILE(RIFR) FROM (RIFREC);
   END FILL;

END QUAL;
```

```
*
*   PROJECT:          REPORT GENERATOR
*   PROGRAMMER:       DEPARTMENT OF COMPUTER SCIENCE
*                     UNIVERSITY OF ILLINOIS
*
*   FILENAME:         LINKFRONTEND.EXEC
*   LANGUAGE:         CMS EXEC
*   DEPENDENCIES:     INITTHESAURUS.PLI          INITREPORTDFNS.PLI
*                     SETDBINTERFACE.PLI         QUERYPARSER.PLI
*                     DECISIONMAKER.PLI
*
*   DATE:             DECEMBER, 1984
*
*
*
*   THIS FILE LINKS THE MODULES COMPRISING THE FRONT
*   END OF THE STANDARD REPORT GENERATOR.
*
GLOBAL T PLILIB
GLOBAL MACLIB DMPLI
LINKEDIT FRONTENDDRIVER INITTHESAURUS SETDBINTERFACE INITREPORTDEFNS QUERYPARSER DEC
```

```
*
*   PROJECT:          REPORT GENERATOR
*   PROGRAMMER:       DEPARTMENT OF COMPUTER SCIENCE
*                     UNIVERSITY OF ILLINOIS
*
*   FILENAME:         STANDARDREPORTGEN.EXEC
*   LANGUAGE          CMS EXEC
*   DEPENDENCIES:
*
*   DATE:             DECEMBER, 1984
*
*
*   THIS FILE CONTAINS THE APPROPRIATE COMMANDS IN ORDER
*   TO RUN THE STANDARD REPORT GENERATOR.
*
*
&CONTROL OFF
-ONEMORE
 GLOBAL MACLIB DMPLI
 GLOBAL T PLILIB
FILEDEF SYSIN TERMINAL (BLKSIZE 80)
FILEDEF SYSPRINT TERMINAL (BLKSIZE 80)
SET PROMPT U
 FILEDEF SKWDOUT DISK SKWDOUT DATA B (RECFM F LRECL 80)
 FILEDEF SRDEF DISK SRDEF DATA B (RECFM F LRECL 1960)
 FILEDEF RIFR DISK RIFR DATA A (RECFM F LRECL 236)
 FILEDEF RIFD DISK RIFD DATA A (RECFM F LRECL 184)
 FILEDEF URIFR DISK RIFR DATA A (RECFM F LRECL 236)
*DEFINE STORAGE 10K
*IPL CMS
 *&CONTROL OFF
 *DEFINE STORAGE 64K
 *IPL CMS
 SET LDRTBLS 07
  FILEDEF DMIN TERMINAL (BLKSIZE 80)
  FILEDEF DMOUT DISK ZACSOUT DATA B (RECFM F BLKSIZE 80)
  FILEDEF DTEST DISK DTEST INDEX B (XTENT 1000)
 FILEDEF DTESTD DISK DTEST DATA B (XTENT 1000)
 FILEDEF DTESTS DISK DTEST SOURCE B (XTENT 1000)
 FILEDEF DTESTE DISK DTEST RECOVER B (XTENT 1000)
 FILEDEF DTESTJ DISK DTEST LOG B (XTENT 1000)
 FILEDEF DTESTA DISK TMPWORK FILE B (BLKSIZE 640)
LOADMOD DM00
YUSINTNEW
FILEDEF RIFR DISK RIFR DATA A (RECFM F LRECL 236
FILEDEF RIFD DISK RIFD DATA A (RECFM F LRECL 184
FILEDEF INDATA DISK INDATA DATA A
* GENERATE REPORT
DISKIO READ RIFR DATA A (ITEM 1 QUIET EOF)
&IF &RETCODE NE 0 &GOTO -PASS
&READ ARGS
&IF &1 LE 0 &GOTO -PASS
QUAL1
&STACK CERLAPP1
&STACK EX CTL
SET ABENDXIT HLIABT
FTFOC2
SET ABENDXIT OFF
ERASE RIFR DATA A
ERASE RIFD DATA A
ERASE INDATA DATA A
-PASS
&BEGTYPE
SELECT ACTION TO BE TAKEN:

        1.  EXECUTE A DIFFERENT QUERY.
        2.  EXIT FROM THE REPORT GENERATOR.
&END
&SPACE 1
&READ VARS &A1
&IF &A1 EQ 1 &GOTO -ONEMORE
&IF &A1 EQ 2 &GOTO -ENDREP
&BEGTYPE
ILLIGAL INPUT        TRY AGAIN
&END
&GOTO -PASS
-ENDREP
&EXIT
```

ACRONYMS

CAPCES:	Construction, Appropriations, Programming, Control, and Execution Systems
CMS:	Conversational Monitoring System
DDD:	Dictionary of Data Definitions
DDS:	Data Dictionary/Directory System
DM:	Decision Maker
DSI:	Data Systems Interface
DTMS:	Data Traffic Management System
FOCUS:	A database management system
HQUSACE:	Headquarters, U.S. Army Corps of Engineers
MCPRS:	Military and Civil Progress Reporting System
MILCON:	U.S. Army Military Construction
OMA:	Operations and Maintenance, Army
PAX:	Programming, Administration, and eXecution
QP:	Query Parser
RAMP:	Responsiveness Analysis of Military Programs
RGF:	Report Generation Facility
RIF:	Report Information File
SRDEF:	Standard Report Definition File
TSK:	Thesaurus of Search Keywords and Internal Keywords
UI:	User Interface
U of I:	University of Illinois
USA-CERL:	U.S. Army Construction Engineering Research Laboratory

USA-CERL DISTRIBUTION

Chief of Engineers
ATTN: Tech Monitor
ATTN: CEIM-SL (2)
ATTN: CECC-P
ATTN: CECW
ATTN: CECW-0
ATTN: CECW-P
ATTN: CEEC
ATTN: CEEC-C
ATTN: CEEC-E
ATTN: CERD
ATTN: CERD-C
ATTN: CERD-M
ATTN: CERM
ATTN: DAEN-ZCE
ATTN: DAEN-ZCF
ATTN: DAEN-ZCI
ATTN: DAEN-ZCM
ATTN: DAEN-ZCZ

FESA, ATTN: Library 22060
ATTN: DET III 79906

US Army Engineer Districts
ATTN: Library (41)

US Army Engineer Divisions
ATTN: Library (14)

US Army Europe
AEAEN-ODCS/Engr 09403
ISAE 09081
V Corps
ATTN: DEH (11)
VII Corps
ATTN: DEH (15)
21st Support Command
ATTN: DEH (12)
USA Berlin
ATTN: DEH (12)
USASETAF
ATTN: DEH (10)
Allied Command Europe (ACE)
ATTN: DEH (3)

8th USA, Korea (19)

ROK/US Combined Forces Command 96301
ATTN: EUSA-HHC-CFC/Engr

USA Japan (USARJ)
ATTN: AJEN-DEH 96343
ATTN: DEH-Honshu 96343
ATTN: DEH-Okinawa 96331

416th Engineer Command 60623
ATTN: Facilities Engineer

US Military Academy 10966
ATTN: Facilities Engineer
ATTN: Dept of Geography & Computer Science
ATTN: DSCPER/MAEN A

AMMRC, ATTN DRXMR-WE 02172

USA AMCCOM 61299-6000
ATTN: AMSMC-RI
ATTN: AMSMC-IS

AMC - Dir., Inst., & Servc
ATTN: DEH (23)
ATTN: AMCEN-A

DLA ATTN: DLA-WI 22314

DNA ATTN: NADS 20305

FORSCOM
FORSCOM Engr, ATTN: AFEN-DEH
ATTN: DEH (23)

HSC
ATTN: HSLO-F 78234
ATTN: Facilities Engineer
Fitzsimons AMC 80240
Walter Reed AMC 20012

INSCOM - Ch, Instl. Div
ATTN: Facilities Engineer (3)

MDW, ATTN: DEH (3)

MTMC
ATTN: MTMC-SA 20315
ATTN: Facilities Engineer (3)

NARADCOM, ATTN: DRDNA-F 01760

TARCOM, Fac. Div. 48090

TRADOC
HQ, TRADOC, ATTN: ATEN-DEH
ATTN: DEH (19)

TSARCOM, ATTN: STSAS-F 63120

USACC, ATTN: Facilities Engr (2)

WESTCOM
ATTN: DEH, Ft. Shafter 96858
ATTN: APEN-IM

SHAPE 09055
ATTN: Surv. Section, CCB-OPS
Infrastructure Branch, LANDA

HQ USEUCOM 09128
ATTN: ECJ 4/7-LOE

FORT BELVOIR, VA 22060 (7)
ATTN: Canadian Liaison Officer
ATTN: British Liaison Officer
ATTN: Australian Liaison Officer
ATTN: French Liaison Officer
ATTN: German Liaison Officer
ATTN: Water Resources Support Ctr
ATTN: Engr Studies Center
ATTN: Engr Topographic Lab.
ATTN: ATZA-DTE-SU
ATTN: ATZA-DTE-EM
ATTN: R&D Command

CRREL, ATTN: Library 03755

WES, ATTN: Library 39180

HQ, XVIII Airborn Corps and Fort Bragg
ATTN: AFZA-FE-EE 28307

Area Engineer, AEDC-Area Office
Arnold Air Force Station, TN 37389

Chanute AFB, IL 61868
3345 CES/DE, Stop 27

Norton AFB, CA 92409
ATTN: AFRCE-MX/DEE

AFESC, Tyndall AFB, FL 32403

NAVFAC
ATTN: Engineering Command (7)
ATTN: Division Offices (6)
ATTN: Naval Public Works Center (9)
ATTN: Naval Civil Engr Lab. (3)

NCEL
ATTN: Library, Code L08A 93043

Defense Technical Info. Center 22314
ATTN: DDA (2)

SETAF Engineer Design Office 09019

Engr Societies Library, NY 10017

Natl Guard Bureau Instl. Div 20310

US Govt Print Office 22304
Receiving Sect/Depository Copies (2)

US Army Env. Hygiene Agency
ATTN: HSHB-E 21010

National Bureau of Standards 20899

RAMP DISTRIBUTION

Chief of Engineers
ATTN: DAEN-ZC
ATTN: DAEN-ZCP
ATTN: DAEN-ZCP-P
ATTN: DAEN-ZCP-R
ATTN: DAEN-ZCP-U

US Army Europe
ATTN: EUDDE 09757
ATTN: EUDCD 09757
ATTN: EUDCD-C 09757
ATTN: EUDED 09757
ATTN: EUDED-M 09757
ATTN: EUDED-MA 09757
ATTN: EUDED-MO/EUDED-MP
ATTN: EUDED-MP 09757
ATTN: EUDED-P 09757
ATTN: EUDED-T 09757

US Army Engineer Division
ATTN: HNDDE 25807
ATTN: HNDED-PM/HNDCD-M 25807
ATTN: HNDED-DM 25807
ATTN: MEDED-M 09038
ATTN: MEDDE 09038
ATTN: MEDPM/MEDCP-P 09038
ATTN: MRDDE 68101
ATTN: MRDCO-C 68101
ATTN: MRDED-MRDMO-A 68101
ATTN: NADDE 10007
ATTN: NADED-M/NADCO 10007
ATTN: NPDDE 97208
ATTN: NPDEN/NPDCO 97208
ATTN: ORDDE 45201
ATTN: ORDED-M/ORDCO 45201
ATTN: PODDE 96858
ATTN: PODED-M/PODCO 96858
ATTN: SADDE 30303
ATTN: SADEN-M/SADCO 30303
ATTN: SPDDE 94111
ATTN: SPDED-T/SPDCO
ATTN: SWDZA 75242
ATTN: SWDED-M/SWDCO 75242
ATTN: MRKDE 64106
ATTN: MRKED-M/MRKCO 64106
ATTN: MRDDE 68102
ATTN: MRDED-M/MRDCO 68102
ATTN: NABDE 21203
ATTN: NABEN-M/NABCO 21203
ATTN: NANDE 10278
ATTN: NANEN-M/NANCO-A 10278
ATTN: NADDE 23510
ATTN: NADEN-M/NADCP 23510
ATTN: NPADE 99150
ATTN: NPAEN-PM/NPACO 99150
ATTN: NPSEN-RS/NPSCO 98142
ATTN: NPSDE 98142
ATTN: ORDED-M/ORDCO 45201
ATTN: ORLDE 40201
ATTN: ORLED-M/ORLCD 40201
ATTN: POFDE 96301
ATTN: POFED-M/POFCD 96301
ATTN: POJDE 96343
ATTN: POJED-M/POJCD 96343
ATTN: SAMDE 36628
ATTN: SAMEN-M/SAMCO 36628
ATTN: SASDE 31402
ATTN: SASEN-M/SASCD 31402
ATTN: SPKDE 95814
ATTN: SPLED-B/SPLCD 90053
ATTN: SPKED-M/SPKCO 95814
ATTN: SPLDF 90053
ATTN: SWFDE 76102

US Army Europe
HQ USAEUR and 7th Army
ATTN: AENEN 09403
ATTN: AENEN-CP 09403
ATTN: AENEN-IF 09403
ATTN: AENEN-MT 09403
ATTN: AEAGS-FMD 09403
ATTN: AEAGD-RM 09403

Commander HQ FORSCOM
ATTN: AFEN-COC 30330

Commander HQ TRADOC
ATTN: ATEN-C 23651

END

FILMED

MARCH, 1988

DTIC